Quality of Service in Vehicular Ad Hoc Networks: Methodical Evaluation and Enhancements for ITS-G5

RAPHAEL MARTIN JÜRGEN RIEBL

Faculty of Computing, Engineering and Media

Submitted in partial fulfilment of the requirements
of De Montfort University for the degree of
Doctor of Philosophy

May 2021

Leicester, United Kingdom

in collaboration with

Ingolstadt, Germany

Supervisors

Dr Ali Hilal Al-Bayatti
Dr Francisco J. Aparicio Navarro

DECLARATION

To the best of my knowledge, I confirm that the work in this thesis is my original work undertaken for the degree of Doctor of Philosophy in the Faculty of Computing, Engineering and Media, De Montfort University. I confirm that no material of this thesis has been submitted for any other degree or qualification at any other university.

ABSTRACT

After many formative years, the ad hoc wireless communication between vehicles has become a vehicular technology available in mass production cars in 2020. Vehicles form spontaneous Vehicular Ad Hoc Networks (*VANETs*), which enable communication whenever vehicles are nearby without need for supportive infrastructure. In Europe, this communication is standardised comprehensively as Intelligent Transport Systems in the 5.9 GHz band (*ITS-G5*).

This thesis centres around Quality of Service (*QoS*) in these VANETs based on ITS-G5 technology. Whilst only a few vehicles communicate, radio resources are plenty, and channel congestion is a minor issue. With progressing deployment, congestion control becomes crucial to preserve QoS by preventing high latencies or foiled information dissemination. The developed VANET simulation model, featuring an elaborated ITS-G5 protocol stack, allows investigation of QoS methodically. It also considers the characteristics of ITS-G5 radios such as the signal attenuation in vehicular environments and the capture effect by receivers.

Backed by this simulation model, several enhancements for ITS-G5 are proposed to control congestion reliably and thus ensure QoS for its applications. Modifications at the GeoNetworking (*GN*) protocol prevent massive packet occurrences in a short time and hence congestion. Glow Forwarding is introduced as GN extension to distribute delay-tolerant information. The revised Decentralized Congestion Control (*DCC*) cross-layer supports low-latency transmission of event-triggered, periodic and relayed packets. DCC triggers periodic services and manages a shared duty cycle budget dedicated to packet forwarding for this purpose.

Evaluation in large-scale networks reveals that this enhanced ITS-G5 system can reliably reduce the information age of periodically sent messages. The forwarding budget virtually eliminates the starvation of multi-hop packets and still avoids congestion caused by excessive forwarding. The presented enhancements thus pave the way to scale up VANETs for wide-spread deployment and future applications.

ACKNOWLEDGEMENTS

Conducting the research work for this thesis has been supported by two universities in collaboration, De Montfort University (DMU) and Technische Hochschule Ingolstadt (THI). I was thus able to draw on advice from my supervisors in England and Germany even-handedly. Throughout the years, Ali Al-Bayatti (DMU) and Christian Facchi (THI) have been reliable constants. Their advice has shaped my academic work to the better considerably, thanks a lot!

At THI, I am happy to see that the group of fellow Vehicle-to-Anything (*V2X*) researchers has grown significantly over the past years. Tinkering with ideas in this open-minded environment has always been a pleasure.

From my former colleagues at THI, I want to point out Andreas Hübner and Martin Bornschlegl thankfully. They have introduced me to the joyful side of academia. Thank you so much for the great time at 'Grillzentrale'!

The AUDI AG, which funded the first years of my research, provided me with valuable first-hand insights into the automotive industry. I am especially grateful to my mentor at AUDI, Sebastian Engel, for his dedication to let me participate in the professional development of V2X systems.

Lucky circumstances brought me together with Hendrik-Jörn Günther, a fellow PhD candidate at Volkswagen Group Research back then. Without him, my V2X simulations would not have led to Artery, and I would miss a good friend.

The developed software accompanying this thesis stands on the shoulders of giants, which I gratefully acknowledge. Numerous building blocks make the foundation for such a complex system, as the bibliography can tell. Nevertheless, I want to point out the creators of OMNeT++ and its INET framework, on which Artery builds upon. Furthermore, thanks to the DLR (German Aerospace Center) and the community behind the traffic simulator SUMO, that keeps the vehicles in my V2X simulations running.

Last but not least, I appreciate the perpetual encouragement and indispensable diversion by my family and friends. Namely, I want to thank my parents, Christine and Jürgen, my sister Rebekka, my fiancée Christina and her parents, Heidi and Georg. Christina is my tower of strength, emotionally and scientifically, in the past and for many years to come. I love you.

Raphael Riebl, December 2020

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

Symbols

3GPP	3rd Generation Partnership Project
5GAA	5G Automotive Association

A

AC	Access Category
ACC	Adaptive Cruise Control
ACM	Association for Computing Machinery
ADAS	Advanced Driver Assistance System
AID	Application Identifier
AIFS	Arbitration Interframe Space
AIMD	Additive Increase Multiplicative Decrease
API	Application Programming Interface
AQM	Active Queue Management
ASN.1	Abstract Syntax Notation One

B

BPSK	Binary Phase Shift Keying
BSM	Basic Safety Message
BSP	Basic System Profile
BSS	Basic Service Set
BTP	Basic Transport Protocol

C

C-ITS	Cooperative Intelligent Transport System
C-V2X	Cellular Vehicle-to-X
C2C-CC	CAR 2 CAR Communication Consortium
CA	Cooperative Awareness
CAM	Cooperative Awareness Message
CBF	Contention-Based Forwarding
CBR	Channel Busy Ratio
CCA	Clear Channel Assessment
CCH	Control Channel
CP	Collective Perception
CPM	Collective Perception Message
CSMA	Carrier Sense Multiple Access
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance

D

DCC	Decentralized Congestion Control
DCC-MCO	DCC Multi-Channel Operations
DCC_ACC	DCC sub-entity at access layer
DCC_FAC	DCC sub-entity at facilities layer

DCC_NET DCC sub-entity at network layer
DEN Decentralized Environmental Notification
DENM Decentralized Environmental Notification Message
DiffServ Differentiated Services
DLL Data Link Layer
DLR German Aerospace Center
DP DCC Profile
DPD Duplicate Packet Detection
DPL Duplicate Packet List
DSRC Dedicated Short Range Communication

E

ECDSA Elliptic Curve Digital Signature Algorithm
EDCA Enhanced Distributed Coordination Access
EEBL Electronic Emergency Brake Light
EPV Ego Position Vector
ETSI European Telecommunications Standards Institute
EU European Union

F

FCC Federal Communications Commission
FIFO First-In-First-Out

G

GAC GeoAnycast
GBC GeoBroadcast
GF Greedy Forwarding
GN GeoNetworking
GNSS Global Navigation Satellite System
GPC GNSS Positioning Correction
GUC GeoUnicast

H

HCF Hybrid Coordination Function

I

ICRW Intersection Collision Risk Warning
IEEE Institute of Electrical and Electronics Engineers
IntServ Integrated Services
IP Internet Protocol
IRT Inter-Reception Time
ISO International Organization for Standardization
ITS Intelligent Transport Systems
ITS-G5 Intelligent Transport Systems in the 5.9 GHz band
IVC Inter-Vehicle Communication
IVI Infrastructure to Vehicle Information
IVIM Infrastructure to Vehicle Information Message

K

KAF Keep-Alive Forwarding
KDE Kernel Density Estimation
KPI Key Performance Indicator

L

LCRW Longitudinal Collision Risk Warning
LocT Location Table
LocTE Location Table Entry
LOS Line-of-Sight
LTE Long-Term Evolution

M

MAC Medium Access Control
MANET Mobile Ad Hoc Network
MAPEM Map Topology Extended Message
MCO Multi-channel Operation
MCS Modulation and Coding Scheme
MFR Most Forward within Radius
MTU Maximum Transmission Unit

N

NAV Network Allocation Vector
NIC Network Interface Card
NLOS Non-Line-of-Sight
NLOSb Non-Line-of-Sight due to obstructions by buildings
NLOSf Non-Line-of-Sight due to obstructions by foliage
NLOSv Non-Line-of-Sight due to obstructions by vehicles

O

OCB Outside the Context of a Basic Service Set
OFDM Orthogonal Frequency Division Multiplexing
OSI Open Systems Interconnection
OSM OpenStreetMap

P

PDR Packet Delivery Ratio
PER Packet Error Rate
PHY Physical Layer
PV Position Vector

Q

QAM Quadrature Amplitude Modulation
QoE Quality of Experience
QoS Quality of Service
QPSK Quadrature Phase Shift Keying

R

RHS Road Hazard Signalling

RLT	Road Lane Topology
RSSI	Received Signal Strength Indicator
RSU	Road-side Unit
RTCM	Radio Technical Commission for Maritime Services

S

SCF	Store-Carry-Forward
SDR	Software-Defined Radio
SHB	Single Hop Broadcast
SNIR	Signal to Noise plus Interference Ratio
SPATEM	Signal Phase and Timing Extended Message

T

TAI	International Atomic Time
TC	Traffic Class
TC-ID	Traffic Class Identifier
TCP	Transmission Control Protocol
TDC	Transmit Data rate Control
TLM	Traffic Light Manoeuvre
TPC	Transmit Power Control
TraCI	Traffic Command Interface
TRC	Transmit Rate Control
TSB	Topologically Scoped Broadcast
TTC	Time To Collision

U

UDP	User Datagram Protocol
UTC	Coordinated Universal Time

V

V2X	Vehicle-to-Anything
VANET	Vehicular Ad Hoc Network

W

WLAN	Wireless Local Area Network

1 INTRODUCTION

Inter-Vehicle Communication (*IVC*) aims to improve safety, comfort and efficiency in road traffic [53]. As a first step, vehicles are now equipped with IVC applications that increase mutual awareness and can exchange warnings about various road hazards [164]. Mutual awareness is achieved by sending so-called beacons via IVC in periodic intervals so traffic participants can recognise each other wirelessly even when out-of-sight. Furthermore, when a vehicle detects a dangerous situation, it can warn peer vehicles to take appropriate actions, *e.g.* slowing down carefully in advance. Typical examples for dangerous situations, which can be detected by vehicles' onboard units autonomously, are road works, traffic jams, slippery road surfaces, aquaplaning, or hard braking actions. Beyond exchanging vehicle states and traffic conditions, future driver assistance systems will also share their sensor information and intentions. Vehicles benefit then from an enriched perception of their environment and can coordinate their automated driving manoeuvres. Communication is not limited to vehicles, though: Traffic lights can advise optimal speeds for green waves, and road-side sensors can provide a supplemental bird-view perspective.

One possibility to establish IVC is with the aid of so-called Vehicular Ad Hoc Networks (*VANETs*), enabling road users to communicate directly without requiring any network infrastructure. Depending on the particular IVC application building upon a VANET, this network type has to fulfil certain performance requirements. For example, many IVC messages have a limited validity duration and must be disseminated in time even when the network is on load. Supporting smooth and steady operations of already specified as well as visionary IVC applications motivates this thesis. Since this work is carried out in Europe, it puts emphasis on Intelligent Transport Systems in the 5.9 GHz band (*ITS-G5*) which represents the European realisation of a VANET. ITS-G5 builds upon customised Wireless Local Area Network (*WLAN*) radios operating in the eponymous 5.9 GHz spectrum.

1.1 PROBLEM STATEMENT

Many concepts and algorithms have been proposed in the past to fulfil IVC's performance needs. From previous work, however, it is already known that *applicability of these proposals is often limited*. Some of them are managing

channel capacity too restrictively and thus will not scale for large networks. Others are unable to treat non-uniform network nodes fairly. These will fail when nodes with varying behaviours are deployed, *e.g.* with different types and revisions of applications.

Besides those known deficiencies, it is also *hardly possible to compare the outcomes of individual performance studies* due to their limited scope and differing evaluation contexts. Systems such as ITS-G5 are inherently complex due to their number of protocol layers and interweaved entities. Thus, publications usually compare only a small sub-set in a few selected traffic scenarios or for single specific use cases.

All in all, a methodology to benchmark competing Quality of Service (*QoS*) mechanisms in VANETs holistically is lacking. Without such a methodical evaluation and subsequent system improvements, the deployment of a future-proof ITS-G5 network is fraught with problems. Quite likely, future applications will demand even more challenging QoS properties and thus aggravate those existing issues.

1.2 OBJECTIVES

The research presented in this thesis aims to improve the communication performance of IVC applications, *i.e.* raising the QoS offered by VANETs employing the ITS-G5 protocol stack. Ideally, a VANET scales flawlessly from sparse to dense network topologies. Both extremes have their unique problems, though: Sparse networks have to cope with recurring, disconnected links among network nodes, which makes message dissemination challenging. In dense networks, the channel capacity becomes the limiting factor because the shared radio medium shall not be overloaded. With the demand for scalability and reliable message dissemination in mind, the following questions emerge :

- How to assess the performance of a VANET methodically?
- Which performance deficits exist in the ITS-G5 protocols?
- What can be done to tackle these issues?
- Which effect do the proposed measures show?

In the course of finding answers to these questions linked to the stated aim, the present thesis pursues the following objectives:

Analysis of the state of the art and the QoS requirements in ITS-G5

At first, the various standardised entities and protocols of an ITS-G5 system are reviewed to gain insights into its communication capabilities. Afterwards, the requirements of typical Intelligent Transport Systems (*ITS*) use cases towards the communication network need to be analysed.

Identification of QoS bottlenecks and proposal of eligible enhancements

The communication patterns employed by the ITS applications influences QoS decisively: Aggressive dissemination of messages by one application can severely congest the network and thus hamper another application's operation. It is the communication system's task to support all applications' demands as good as possible. Hence, the entire ITS-G5 network stack has to be examined in terms of potential bottlenecks. The found weaknesses, especially those limiting QoS, are addressed by proposing enhancements fitting into the existing architecture.

Design and realisation of a complete ITS-G5 simulation environment

Another objective of this thesis is the provision of an elaborate simulation environment, which can be employed to evaluate ITS-G5 networks without neglecting significant aspects. In particular, the provided simulation model strives to be compliant with the standardisation. Furthermore, heterogeneity in terms of vehicles' applications and road traffic is to be supported to avoid optimisation towards a single use case or environment.

Evaluation of enhancements by means of simulation studies

With such a simulation environment in place, which can handle the full complexity of ITS-G5 systems, QoS will be evaluated within sufficiently realistic environments. Especially the impact of the identified ITS-G5 weak points and their proposed countermeasures can be studied thoroughly.

In a nutshell, this thesis aims to forge a bridge from the analysis of QoS requirements to eliminating problems in the current ITS-G5 system design and the methodical evaluation of VANET communication performance. With this clear focus on performance, *i.e.* non-functional requirements of IVC communication, functional safety aspects are out of scope. For those readers interested specifically in functional safety aspects of IVC use cases, Autotalks [192] may be a good starting point. As soon as systems employing IVC not only alert drivers but also step into vehicle controls, management of

functional safety as outlined by *ISO 26262* [146] becomes mandatory. For the remainder of this thesis, though, IVC units are expected to adhere to the standardised specifications. Any malfunctions are equally neglected, regardless of whether they are due to possible assembly errors, accidental damages or hardware faults in processors, memory, and radio equipment. However, this does not preclude any flaws related to the IVC specifications or demanding conditions for wireless communications, such as interference or short capacities.

1.3 THESIS OUTLINE

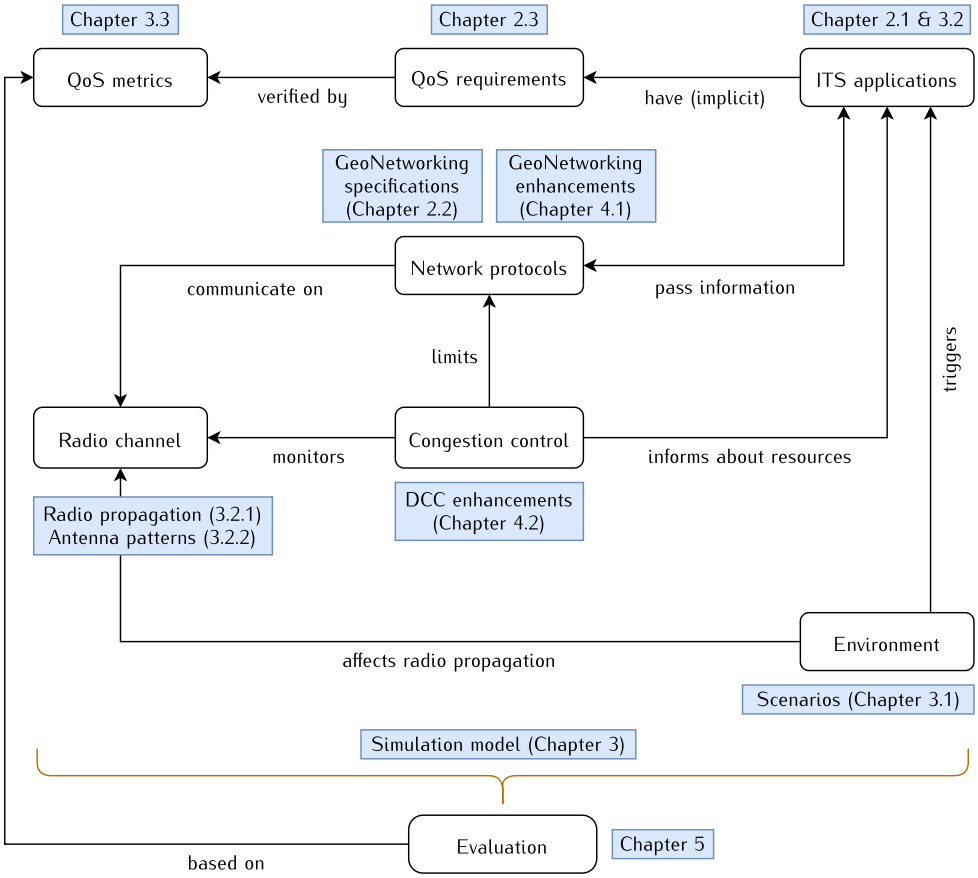


Figure 1.1: Major components of this thesis and their context

The diagram shown in Fig. 1.1 relates the individual subtopics of this thesis to one another. The starting point is the ITS applications, which have expectations towards the QoS offered by the network. Explicit QoS requirements are hardly found in standardisation, though. Hence, QoS as known from related disciplines and wireless networks is also reviewed.

The performance of ITS applications is an interplay of the operational environment, employed network protocols, and congestion control. Environmental stimuli trigger applications in most cases, either by detecting events

in road traffic or just the hosting vehicle's state. The environment also affects the radio propagation and thus the communication range of applications. Furthermore, those stations within range have also to deal with their mutually induced channel congestion. Chapter 2 reviews the state of the art and related work regarding ITS applications, communication protocols, congestion control and QoS.

1.3.1 Methodology for Evaluating QoS in VANETs

Chapter 3 contributes a concept to evaluate QoS-related algorithms and parameterisation of VANETs methodically. This concept directly approaches the stated problem of lacking comparability between previous studies due to varying fundamental assumptions, communication stack configurations and evaluation scenarios. The key element of the conceived concept is a QoS test bench employing state-of-the-art simulation tools.

The elaborated simulation environment strives to capture all major influential variables with respect to communication performance in ITS-G5 VANETs. Hence, the existing tools are considerably extended by a full ITS-G5 protocol stack implementation, and several simulation features not found elsewhere. The resulting simulation environment comprises detailed models of applications, protocols, radio propagation and congestion. Particular emphasis is put on modelling radio congestion efficiently, *i.e.* with reduced computational costs.

Furthermore, typical scenarios for ITS-G5 communication are elaborated and shared with the research community for future studies. Ultimately, this comprehensive QoS test bench enables VANET evaluations in a controlled and reproducible environment.

1.3.2 DCC and GeoNetworking Enhancements

An inherent characteristic of ITS-G5 is its necessity to manage channel congestion. If channel congestion increases boundlessly, the communication performance degrades unavoidably. The importance of congestion control in ITS-G5 is emphasised by the existence of a dedicated entity for this purpose, named Decentralized Congestion Control (*DCC*). Furthermore, inadequate behaviour by ITS-G5's network layer can cause excessive channel usage because of buffered or duplicated packets. Hence, the network protocols and congestion control mechanisms are not just studied as defined in ITS standards and research, but specific improvements are proposed in Chapter 4.

These enhancements include but are not limited to the monitoring of channel load, resource planning in terms of channel occupancy, and the efficient collaborative dissemination of messages. While various congestion

control algorithms assume single or at least uniform application deployments, the proposed enhancements take the applications' diversity in communication patterns into account. The unifying objective of these enhancements is to increase the amount of information that can be conveyed in time between vehicles. Because data is predominantly addressed to vehicle groups, the rule of thumb is: The more packets can be received successfully and swiftly, the better the QoS for ITS applications.

1.3.3 Evaluation of QoS Mechanisms

Once the suggested ITS-G5 improvements are laid out, their evaluation follows in Chapter 5. These evaluations make use of the introduced simulation environment acting as a QoS test bench. Several VANETs configurations allow comparing the state-of-the-art VANETs, proposals by third parties and enhancements outlined herein. This study is multidimensional with respect to employed road topologies (urban, rural, highway), traffic demands, routing features, congestion control mechanisms, queuing, and IVC application loads. The starting point for each configuration is the Basic System Profile (*BSP*) created by the CAR 2 CAR Communication Consortium (*C2C-CC*) [163], building upon the European Telecommunications Standards Institute (*ETSI*) standards. Those configurations are then compared by carefully selected QoS metrics, where the BSP constitutes the baseline for the currently achievable performance. This evaluation shows the features' effects on IVC performance in isolation as well as their side effects in combination.

This thesis is vigilant about the interweaving of the diverse components in ITS-G5 systems and thus approaches QoS in ITS-G5 networks holistically. One question spans all parts: How to improve the performance of ITS-G5 VANETs?

2 STATE OF THE ART AND RELATED WORK

This chapter recapitulates publications related to QoS in ITS-G5 VANETs. Obviously, the standardisation documents by ETSI set the baseline for the performance achievable by ITS-G5 stations as of today. Section 2.1 offers an overview of ITS-G5's architecture and its various layers. In a VANET based on ITS-G5, messages can be disseminated collaboratively through multi-hop packet forwarding. These forwarding algorithms and related packet repetition strategies, which help to spread information over space and time, are discussed in Section 2.2. Section 2.3 addresses QoS from various perspectives. This review includes QoS as found in Internet Protocol (*IP*) networks, specific QoS challenges found in wireless networks and QoS requirements of IVC. As it turns out, QoS is closely related to controlling congestion in the wireless medium. Various congestion control mechanisms are highlighted in Section 2.4 and how some of them have influenced standardisation.

2.1 STANDARDISATION OF ITS-G5

ETSI drives standardisation of ITS-G5 forward and is supported by experts from industry and academia. Groups such as the C2C-CC are not setting standards on their own but provide input to ETSI and also politics. The European Commission's Delegated Act on Cooperative Intelligent Transport System (*C-ITS*) [177] bears the hallmarks of C2C-CC with respect to the arrangement of the cited ITS specifications by ETSI and other standardisation bodies, *e.g.* SAE International and International Organization for Standardization (*ISO*).

Specifications by ETSI often allow some tailoring to particular use cases. For example, the GeoNetworking (*GN*) specification explicitly defines a set of protocol constants [193, Annex H] which shall be fixed in a specific deployment but can be adjusted on a case-by-case basis otherwise. The C2C-CC thus manages and releases another set of documents centred around their BSP [163] that further prescribe how ETSI ITS specifications shall be employed for IVC deployments. Whenever 'BSP' is mentioned, this particular document is meant, which has also been a blueprint for the C-ITS Delegated Act in large part [177]. For reference, *TR 101 607* lists all ITS related ETSI documents

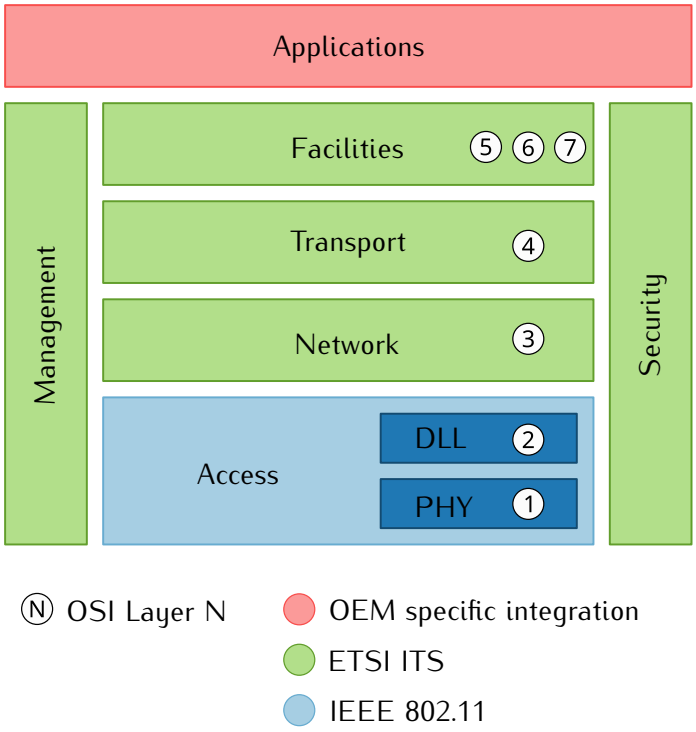


Figure 2.1: Architecture of an ETSI ITS station as per [57]

belonging to 'Release 1' [194]. Though the passing of the aforementioned Delegated Act has been stopped by member states, this draft bill is still a good reference about Europe's idea of a common C-ITS deployment.

A look at the communication architecture gives a good initial overview of an ITS-G5 system [57] and its fundamental components. In principle, the architecture follows the Open Systems Interconnection (*OSI*) model [20] by separating concerns into layers. Figure 2.1 shows this architecture and puts the ITS layers in relation to the seven OSI layers. ETSI defines its ITS architecture independent of any particular radio technology in the first place. By employing ad-hoc wireless network technology in the 5.9 GHz band [86], this generic architecture becomes ITS-G5. This access layer specification (blue dyed in Fig. 2.1) relies on Institute of Electrical and Electronics Engineers (*IEEE*) 802.11 [121] at its core. Section 2.1.2 will introduce the access layer in more detail and also its two sub-layers, Physical Layer (*PHY*) and Data Link Layer (*DLL*).

ETSI specifies its own layers (coloured in green) on top of IEEE's access layer. These layers comprise packet routing via GN [193], a Basic Transport Protocol (*BTP*) [170], and a so-called Facilities layer realising the uppermost OSI layers. In ETSI ITS terminology an 'application' refers to a vehicle feature such as an Advanced Driver Assistance System (*ADAS*) using C-ITS information. From a communication stack's perspective, the OSI application layer is located at ITS Facilities, though. It is the vehicle manufacturer's

decision how to make use of C-ITS information at other vehicle components and then also depending on the particular vehicle platform. Thus, ETSI cannot specify this ITS application layer (red-dyed) at all.

The architecture further includes two cross-layers, 'management' and 'security'. Former layer is meant to make configuration parameters and system states available across the whole ITS stack. Security, the latter layer, penetrates the stack at various points to enforce specific security aspects at the respective layers. Noteworthy, the original architecture diagram from *EN 302 665* does not explicitly mention the DCC cross-layer, which is nevertheless a mandatory component of every ITS-G5 station.

2.1.1 Use Cases and Applications

ITS use cases are mainly motivated by the intent to increase road safety and traffic efficiency by direct communication among road participants [53]. In support of these aims, the European Union designated a dedicated frequency band at 5.9 GHz to be used for these purposes, as mentioned earlier [103]. One characteristic of applications in the traffic domain is that exchanged information is only relevant for an ad-hoc group of other vehicles within a close geographic area, e.g. speed advisories at traffic lights. Another characteristic is the often short-lived validity of exchanged information, such as pre-crash warnings.

Other use cases related to infotainment, fleet management, software updates etc. are already realised or can be realised based on IP communication over cellular networks because they are neither short-lived nor narrow-scoped. This thesis considers only use cases relying on ad-hoc communication.

According to [177, Annex 2, 2.8], a vehicle capable of C-ITS communication shall operate a Cooperative Awareness (*CA*) and Decentralized Environmental Notification (*DEN*) service. The CA service requires the vehicle to transmit a beacon periodically, at least once per second. Other road users can thus perceive the presence of this vehicle solely based on wireless communication even if the line-of-sight is blocked. Another advantage over conventional sensors is that Cooperative Awareness Messages (*CAMs*) also convey information that is otherwise hard or impossible to gather, e.g. if a vehicle transports dangerous goods. While CAMs are sent periodically, a Decentralized Environmental Notification Message (*DENM*) transmission is always triggered by an associated event.

Roadside units are expected to support specific infrastructure services, namely Infrastructure to Vehicle Information (*IVI*), Road Lane Topology (*RLT*), and Traffic Light Manoeuvre (*TLM*) [177].

Table 2.1: Frequency and channel allocation for ITS in the EU

Band	Usage	Name	Centre frequency	Channel number
ITS-G5A	road safety	G5-CCH	5900 MHz	180
		G5-SCH1	5880 MHz	176
		G5-SCH2	5890 MHz	178
ITS-G5B	non-safety	G5-SCH3	5870 MHz	174
		G5-SCH4	5860 MHz	172
ITS-G5D	future use	G5-SCH5	5910 MHz	182
		G5-SCH6	5920 MHz	184

Industry and academia also spend much attention to use cases designed for deployment after basic CA and DEN services. The truck industry is keen on platooning where vehicles reduce the gaps between them for better fuel economy. As those gaps are becoming too small for safe operation by human drivers, management and control messages exchanged via IVC help to organise platoons and keep platoon members at safe distances [138]. Collective Perception (*CP*) serves to share information about objects detected by a vehicle's sensors. Standardisation of collective perception is still ongoing but *TR 103 562* includes already a thriven proposal for the Collective Perception Message (*CPM*) format [173]. Thinking about the distant future, IVC will contribute to coordinate automated driving manoeuvres [164]. At first, the semi-automated cooperative Adaptive Cruise Control (*ACC*) is a step in this direction, where an evolved CA service shall enhance control of a safe longitudinal gap between vehicles [174].

2.1.2 Access Layer

At the lowest layer, ITS-G5 signals comply with the Orthogonal Frequency Division Multiplexing (*OFDM*) PHY system specification of IEEE 802.11 [121, Chapter 17] occupying 10 MHz channel bandwidth. This channel spacing allows for data rates between 3 Mbit/s to 27 Mbit/s. Three particular rates – 3, 6, and 12 Mbit/s – have to be supported by every station.

On the territory of the European Union (*EU*), a dedicated frequency spectrum is allocated for ITS applications [86, Table 1]. This spectrum ranges from 5.855 GHz to 5.925 GHz and is divided into sub-bands and channels, as shown in Table 2.1. A similar frequency spectrum at 5.9 GHz has originally been designated for ITS usage in the United States of America [25]. Meanwhile, this allocation for Dedicated Short Range Communication (*DSRC*) has been revoked by the Federal Communications Commission (*FCC*) at the end of 2020 [206]. As far as anticipated deployment in Europe is concerned, only the Control Channel (*CCH*) is going to be used initially. Communication in the *G5-CCH* channel employs 6 Mbit/s as the default data rate.

The introduction of the Outside the Context of a Basic Service Set (*OCB*) mode in IEEE 802.11 is the most significant change compared to other WLAN variants. This mode disables the otherwise mandatory organisation of network participants in Basic Service Sets (*BSSs*). In a VANET, however, this organisation is not applicable due to rapidly changing network topologies. Highly mobile network nodes such as vehicles would waste much time for establishing short-lived BSSs. Hence, a fixed common BSS 'wildcard' identifier is used for every VANET participant. In OCB mode any station is allowed to transmit data frames without prior hand-shaking procedure. As no authentication and data confidentiality can be provided by 802.11 without a BSS, those security aspects need to be handled by upper layers in VANETs.

The access layer's DLL employs the IEEE 802.11 Medium Access Control (*MAC*) to coordinate shared usage of the radio medium. Access to the medium follows the Carrier Sense Multiple Access with Collision Avoidance (*CSMA/CA*) principle, i.e. any station listens to the channel before transmitting anything. Only if the channel is sensed to be free, a transmission can be started. Due to the prevalent half-duplex nature of current radio technology, a transceiver can either receive or transmit at the same time. Thus, it cannot detect if another station accidentally starts transmitting during its own transmission process. Their transmissions will thus mutually interfere and adversely affect the signal quality for all receivers. In many cases, the resulting Signal to Noise plus Interference Ratio (*SNIR*) will be too low for successfully decoding either transmission, resulting in a packet collision. Such collisions on the radio medium are thus to be avoided as much as possible.

The collision avoidance of CSMA/CA aims at reducing the risk for concurrent transmission starts by adding a backoff procedure. When the radio medium becomes free, a station will not start its transmission immediately but will defer it by a random amount of time. In 802.11, this time is always a multiple of the slot time, which is 13 µs in 10 MHz channels [121, Table 17-21].

Beside CSMA/CA procedures, the harmonised European standard *EN 302 571* [128] further restricts when and how transmissions are allowed in ITS-G5 channels. According to this standard, a single transmission is never allowed to occupy the radio medium longer than 4 ms. This implies no restrictions for transmissions at 6 Mbit/s considering the maximum MAC payload length of 2304 B [121, Table 9-19]. However, it limits the use of slower modulation and coding schemes. Furthermore, a station has to wait at least 25 ms before emitting a follow-up transmission. If the overall channel congestion exceeds 62 %, the demanded period of silence is further extended

Table 2.2: Enhanced Distributed Coordination Access (*EDCA*) parameters of ITS-G5 according to [89, Table 4] and [121, Table 9-138]

Access Category		Contention Window		AIFS
		CW_{min}	CW_{max}	
AC_VO	Voice	3	7	58 µs
AC_VI	Video	7	15	71 µs
AC_BE	Best effort	15	1023	110 µs
AC_BK	Background	15	1023	149 µs

up to 1 s, depending on the present congestion level and the duration of a station's previous transmission. Last but not least, transmissions' mean equivalent isotropically radiated power is limited to 33 dBm (about 2 W).

IEEE 802.11 MAC also introduces a QoS mechanism in the DLL capable of prioritising packets. This mechanism is called EDCA [121, p. 10.2.4.2] and mandatory for any ITS station [86, p. 4.6]. According to a packet's priority, it is enqueued in one of four Enhanced Distributed Coordination Access (*EDCA*) queues, each representing one priority class named access category. From lowest to highest priority, these are *background* (AC_BK), *best effort* (AC_BE), *video* (AC_VI), and *voice* (AC_VO). Prioritisation is realised by employing different backoff parameters for each access category queue, as shown in Table 2.2. A queue of higher priority resumes reducing the backoff counter earlier after the medium becomes idle than a lower priority because its assigned Arbitration Interframe Space (*AIFS*) is shorter. Furthermore, the size of the contention favours packets of higher priority. The initial backoff period of a packet is the fixed slot time of 13 µs multiplied by a randomly drawn number of backoff slots varying from 0 to CW_{min}, the shortest contention window. If a unicast transmission fails, *i.e.* the sender gets no acknowledgement from the receiver, each retry doubles the size of the contention window up to CW_{max}. Broadcast transmissions are never acknowledged, and thus only the shortest contention window applies to them. Though higher priorities are handled with preference by these settings, it has to be noted that packets with lower priorities do not necessarily starve in their queues. Along with their waiting time, also their probability for transmission will increase as their remaining backoff duration shrinks. For example, a pending AC_VI broadcast frame may be sent before an AC_VO broadcast frame because AC_VI's minimal backoff (71 µs) is smaller than the largest possible of AC_VO ($58\,\mu s + 3 \cdot 13\,\mu s = 97\,\mu s$). However, an AC_VO broadcast frame will always be sent before a pending AC_BE or even AC_BK frame.

Table 2.3: Available GN packet transport types

Transport type	Destination
SHB Single Hop Broadcast	one-hop neighbours
TSB Topologically Scoped Broadcast	n-hop neighbours
GUC GeoUnicast	one host identified by its GN address
GAC GeoAnycast	at least one station in destination area
GBC GeoBroadcast	all stations in destination area

2.1.3 Network Layer

The GeoNetworking (*GN*) layer [193] is a true workhorse of ITS-G5 VANETs and responsible for several tasks:

- Routing of packets according to various modes
- Optionally buffering packets for later dissemination in sparse networks
- Encapsulating payload as a secured message

GN supports five routing modes as listed in Table 2.3, but only two are of practical relevance. In the terminology of GN, these routing modes are called 'packet transport types'. Single Hop Broadcasts (*SHBs*) are simple broadcasts to all neighbours reachable via a single hop, *i.e.* no forwarding by receivers applies. This mode is typically used for short-range, periodic communication such as CAMs and CPMs. GeoBroadcast (*GBC*) is a considerably more sophisticated transport type as it addresses the group of stations within a given geographical area. The destination area of a GBC can be described as circle, ellipse or rectangle centred at an arbitrary geodetic datum point.

Support of all these features requires a relatively sophisticated header design as depicted in Fig. 2.2. All GN packets start with a *Basic* header, which is the only part that can be modified by relaying stations when forwarding the packet. For one thing, the *Lifetime* field is expected to get reduced by the sojourn time in any GN buffer. For another thing, the forwarding node needs to decrement the number of remaining hops.

Headers following the *Basic* header are usually wrapped into a *Secured Message* header and their contents are cryptographically secured. Hence, forwarding nodes cannot change the content of other headers without breaking the signature enclosed in the *Secured Message*. Receiving stations can verify the validity of the signature if they know the public key used for signing the message. Usually, this public key is shared via the attached certificate or has been shared previously. In the latter case, it is sufficient to refer to the

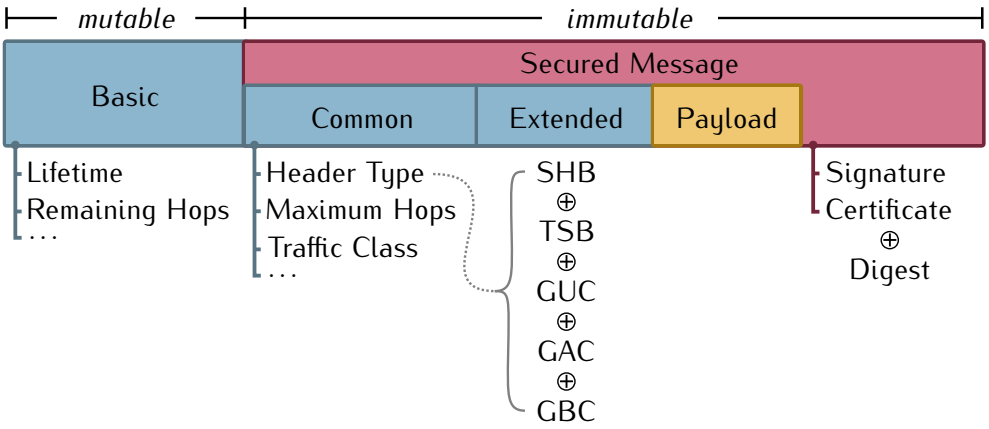


Figure 2.2: GN header structure

full certificate only by its digest. The detailed security operations are not of particular interest in this thesis because a sufficiently dimensioned security module is assumed, *i.e.* the processing delay to sign and verify messages is negligible. The overhead induced by increased message length, however, is fully taken into account by implementing the *Secured Message* format [106].

Each GN packet also includes a Traffic Class (*TC*) field which is set by the source station. This field contains two flags indicating if the packet is allowed to get buffered through Store-Carry-Forward (*SCF*) and or be offloaded to another channel. Additional, it includes a 6-bit wide Traffic Class Identifier (*TC-ID*), whose interpretation depends on the employed medium. Only 4 of 64 identifiers are used for ITS-G5 so far, mapping these to the four access categories of the G5-CCH [195]. The highest priority AC_VO corresponds to TC-ID 0 and the lowest AC_BK to TC-ID 3. Thus, every priority class of EDCA is accessible by upper layers through GN traffic classes. The two highest access categories AC_VO and AC_VI are intended for DENM transmissions by the originating station, whereas forwarded DENMs are only sent with the lowest priority AC_BK [163]. CAMs are associated with TC-ID 2 and thus mapped to AC_BE, i.e. have a higher priority than forwarding traffic but lower than any DENM generated by this station.

While the current state of ITS-G5 specifications about Multi-channel Operation (*MCO*) is still in a state of flux (unspecified behaviour of channel offloading flag and many unused TC-IDs), SCF is specified. Hence, applications are encouraged to set this flag if their packets are delay-tolerant and thus buffering them even for several seconds up to minutes does not invalidate their merit. If the SCF flag is set and a station knows about no direct neighbouring stations, it can buffer the packet up to its remaining packet lifetime. As soon as a neighbour is added to its location table, all buffered packets will be flushed immediately.

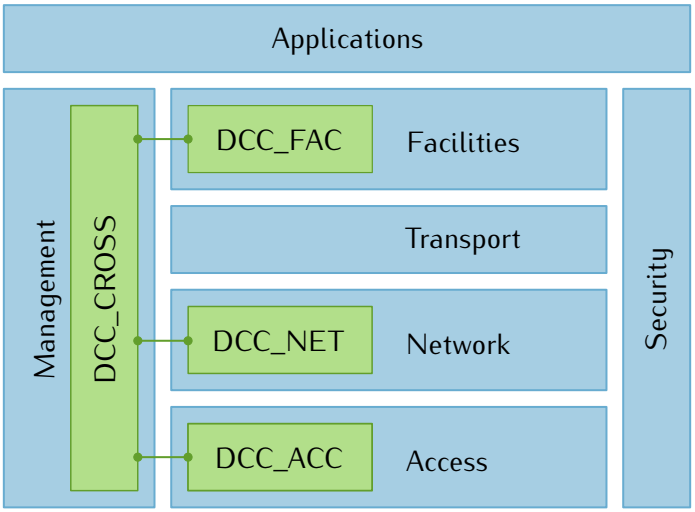


Figure 2.3: DCC architecture of an ETSI ITS station (see [107, Figure 1])

Section 2.2 dissects the GN algorithms with respect to SCF and packet routing. This section also details the bookkeeping in the location table, packet buffer management and forwarding decisions, including several flavours of Contention-Based Forwarding (*CBF*).

2.1.4 Congestion Control

Counter-intuitively, no single entity can be identified as the dominant congestion control entity. Instead, several layers harbour sub-entities that collaborate in controlling network congestion. Their strategies rely on reducing the number of packets, adjusting transmission parameters, measuring the actual congestion level and sharing congestion data with other stations. In terminology of ITS-G5, congestion control is named DCC [86]. Primarily, congestion control is supported by the access, network and facilities layers, which will be re-visited with a focus on congestion control aspects in the following. Figure 2.3 shows the DCC architecture of an ETSI ITS station enhanced by its DCC sub-entities, as outlined in *TS 103 175* [107]. While each sub-entity is embedded into its respective layer, these entities also share information among them via the cross-layer-entity 'DCC_CROSS' acting as a message bus.

Congestion Control at Access Layer

Fundamental limits for communication in the 5.9 GHz ITS band are laid out in European standards [128, 86]. Those have binding character, as they are based on requirements given by the European Parliament and of the Council in Directive 2014/53/EU [98]. DCC at the access layer is the lowest congestion control entity in the protocol stack designed to guarantee the

abidance of those limits [143]. If mechanisms on upper layers fail, this entity is the last line of defence to prevent excessive channel usage by a station. For this reason, DCC at the access layer is often nicknamed 'gatekeeper'.

Measuring the local channel congestion as perceived at a station's antenna is the primary input source for congestion control. The access layer of each station has to report the Channel Busy Ratio (*CBR*) every 100 ms to upper layers for this purpose [128, section 4.2.10]. CBR is defined as the ratio of time the strength of received signals has been above −85 dBm to the whole measuring interval. For the actual measurement procedure, a deviation of 3 % is allowed. Since *EN 302 571* [128] refers only to the strength of received signals, it remains undefined there how own transmissions shall be accounted. The underlying IEEE 802.11 standard, however, says that 'the medium shall be determined to be busy when the STA [station] is transmitting'[121, section 10.3.2.1]. An implementation complying to *EN 302 571* may decide to ignore own transmission durations simply: Since a station's duty cycle, *i.e.* time ratio a radio is transmitting on a channel, is limited to 3 % at most, the reported CBR would be still within the error margin mentioned above. IEEE 802.11 suggests reporting channel load percentage values as linearly scaled integer values from 0 to 255 [121, section 11.11.9.3]. Considering associated discretisation errors (about 0.39 %) and allowed error margins, one should bear in mind that channel load measurements are not very precise. Complying implementations may thus report channel loads varying by some percentage points. Nonetheless, a test case regarding sensitivity threshold correction specified in [107] requires reported CBR values to deviate less than ±1 percentage point.

Since the DCC sub-entity at access layer (*DCC_ACC*) is the last ITS-G5 entity before a to-be-transmitted packet enters the realm of IEEE 802.11, it is naturally in charge of transmission parameters passed on to 802.11 MAC. In particular, it can declare the intended transmission power and data rate. Along with control of the timing when packets are passed on to MAC, three major congestion control techniques can be identified [143]:

Transmit Power Control (*TPC*) DCC is expected to adapt the transmission power [128, section 4.2.4]. The charm of TPC is the range reduction in which a transmission interferes with others. Hence, at an equal distance from the transmitter, a transmission with lower power will contribute less to the measured CBR. However, it is hard to estimate the actual range because signal path loss is profoundly affected by the environment.

TRANSMIT DATA RATE CONTROL (*TDC*) With TDC a station can adjust the duration a packet with a given payload is occupying the channel. Neglecting the short packet preamble (physical layer) with its fixed data rate, switching from 6 Mbit/s to 12 Mbit/s cuts the air duration by half. The downside is a less robust signal modulation which is more prone to interference nearby.

TRANSMIT RATE CONTROL (*TRC*) TRC is the most fundamental gatekeeper mechanism. It either defers or entirely drops to-be-transmitted packets when they arrive from upper layers more frequently than allowed by given limits. Since finding a good trade-off in tuning power and data rate is hard, TRC is the only mechanism covered by standardisation.

Present standardisation documents do not specify TPC nor TDC in great detail, *i.e.* implementors are free to use either fixed values or any arbitrary algorithm, as long as the regulatory constraints are honoured. For example, the maximum transmission power is 33 dBm in general but in the vicinity of existing toll stations operating at 5.8 GHz, further restrictions down to 10 dBm may apply [128]. Definite TPC and TDC algorithms are only covered by research so far, which are detailed in Section 2.4.2. With respect to TRC, ETSI outlines two distinct approaches: The older reactive approach relying on a finite state machine, and the more recent adaptive approach inspired by LIMERIC [63].

REACTIVE DCC_ACC The reactive approach revolves around state transitions triggered by measured CBRs. Each state is then associated with a T_{off} period, *i.e.* the minimum duration between consecutive transmissions by a particular station. Thus, a station using this approach reacts upon increasing congestion by extending its T_{off}. Two mappings are given by TS 102 687, depending on the air duration of packet transmissions [143, Table A.1 and A.2]. Both variants define five states, with a 'relaxed' state when CBR is below 30 % and a 'restrictive' state above 60 % or 65 %, respectively. Between 'relaxed' and 'restrictive', three 'active' states are defined. Transitions are only allowed between neighbouring states, *e.g.* from 'relaxed' to 'active 1'. Even in the most permissive 'relaxed' state, $T_{off} = 50\,ms$ is considerably more conservative than the limit of 25 ms given by EN 302 571.

Superseded version 1.1.1 of DCC_ACC set 'relaxed' as initial state explicitly [61]; however, this information is not given anymore. It is hard to tell if this has been left out on purpose to give implementers more degrees of freedom or has been forgotten. However, one has to consider the fact that the specification of state machines has changed in several aspects from release 1.1.1 [61] to 1.2.1 [143]. For example, the 'active' states were fully meshed in the earlier release, *i.e.* neighbour states could be skipped. Furthermore,

the current release has simplified the reactive approach in many aspects, *e.g.* details regarding TPC and TDC are not given any longer. The time lag in ramping up and down states has been omitted entirely by now as well.

The DCC defined in the *Basic System Profile* [163] builds upon the reactive approach and specifies a state machine, as shown in Fig. 2.4. This state machine reuses mostly the parameterisation suggested by ETSI [143, Table A.2], but C2C-CC refines it in some respects. For one thing, the locally measured CBR values are averaged over the last two measurement intervals. Hence, input to the state machine CBR_{in} is the mean of the current $CBR_{raw}(n)$ and previous $CBR_{raw}(n-1)$ CBR measurements. These raw CBR measurements stem from the station's access layer. For another thing, additional packet bursts are allowed when in 'relaxed' state only. Bursts allow transmission of several packets with the highest priority in a row, only limited by the burst duration of one second and not more than 20 packets per burst. Those bursts may reoccur every 10 seconds. This modification is in line with the *EN 302 571* [128], as lower priority traffic can only sum up to 16.7 messages per second and such a burst can contribute another 20 messages per second at most. Hence, the limit of 40 messages per second ($T_{off} = 25\,ms$) is not exceeded.

ADAPTIVE DCC_ACC The cornerstone of the adaptive approach is the calculation of δ, which is defined as the allowable duty cycle for each station to maintain a global target CBR CBR_{target}. δ is updated every 200 ms with the aim that the smoothed CBR_{local} matches CBR_{target} as closely as possible. The details of adapting δ are discussed in more detail later in Section 2.4.2. In fulfilment of ETSI limits, the δ is capped at 3 %, so the maximum duty cycle is never exceeded. At the same time, a minimum duty cycle of 0.06 % is guaranteed to prevent starvation in highly congested situations.

While the *Basic System Profile* [163] demands to smooth raw CBRs by averaging the last two measurements, *TS 102 687* [143] goes even further for the adaptive approach: The mean of raw measurements is used as update term to a moving average calculation, also taking into account the previously calculated (smoothed) $CBR_{local}(n-2)$ as shown in Eq. (1). Time points of updating CBR_{local} are expected to be synchronised among all stations. This synchronisation relies on Coordinated Universal Time (*UTC*) time, *i.e.* the update calculation is triggered when the UTC timestamp modulo 200 ms is zero. An increment of n translates to 100 ms of elapsed time, the interval at which raw CBR measurements are reported. The time points of these raw measurements are unsynchronised, *i.e.* $CBR_{raw}(n)$ and $CBR_{raw}(n-1)$ refer to the two latest measurements since the previous $CBR_{local}(n-2)$

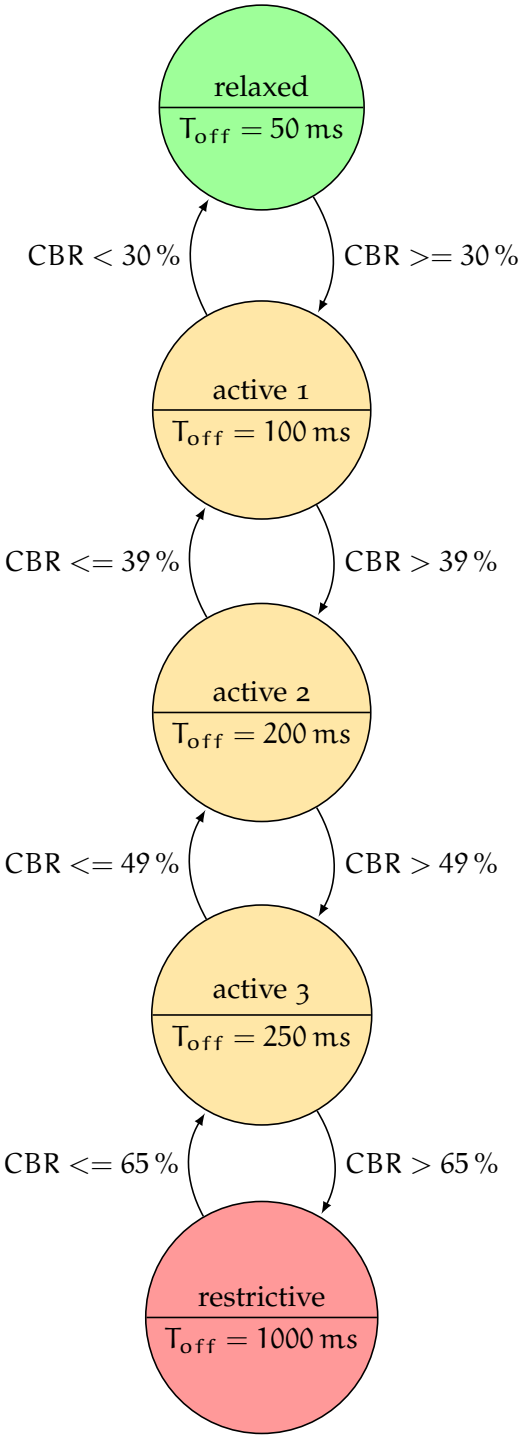


Figure 2.4: Reactive DCC state machine as per *Basic System Profile* [163]. Initial state is not explicitly given.

calculation. Despite possible clock synchronisation via Global Navigation Satellite System (*GNSS*) one may still expect some deviation of calculation time points, though.

$$\begin{aligned} CBR_{local}(n) = 0.5 \cdot CBR_{local}(n-2) \\ + 0.25 \cdot (CBR_{raw}(n) + CBR_{raw}(n-1)) \end{aligned} \quad (1)$$

The potential risk of artificially synchronising transmissions among stations is also reflected in *TS 102 687* [143]. Unsurprisingly, all (nearby) stations determine at the same time their updated δ. With an increasing δ and packets queued for the next transmission opportunity, those stations may schedule immediate transmissions then at these synchronised time points. In order to prevent this problem, a gatekeeper algorithm for the adaptive approach is outlined in [143, Annex B]. At its heart, the waiting period between transmissions T_{off} is controlled by Eq. (2), which is in line with ETSI by clamping $T_{off} \in [25\,ms, 1\,s]$.

$$T_{off} = \min(\max(T_{on}/\delta, 25\,ms), 1\,s) \quad (2)$$

Equation (2) spreads time points of transmissions uniformly in the duty cycle, but only if packets have identical air durations T_{on}. However, even a station only transmitting CAMs will have a variety of air durations because of the presence (or absence) of optional containers in its message format. In order to prevent synchronisation of transmission time points, *i.e.* T_{off} of several stations ending at the same time point, ETSI suggests updating T_{off} by Eq. (3), which is a conversion of [143, equation B.2] and Eq. (2).

$$T_{off} = \min(\max(\frac{T_{on}}{\delta} \cdot \frac{T'_{off,left}}{T'_{off}} + T'_{off,elapsed}, 25\,ms), 1\,s) \quad (3)$$

T'_{off} denotes the waiting period as calculated previously, *i.e.* the original T_{off} calculated by Eq. (2) before the update. At the update time point, $T'_{off,elapsed} + T'_{off,left} = T'_{off}$ denote how much time has passed since the transmission and how much time still remains from the original T'_{off} at the moment. The factor $T'_{off,left}/T'_{off}$ gives those stations an advantage, that have only a minor part of their previous waiting period left. The updated T_{off} is still relative to the last transmission time point, but the next transmission cannot lie in the past obviously. Hence, the duration since the last transmission until now $T'_{off,elapsed}$ is added.

If the update cycles were not synchronised at all, one could avoid this complexity altogether. The pros and cons of synchronised versus unsynchronised update cycles are discussed in more detail in Section 2.4.2.

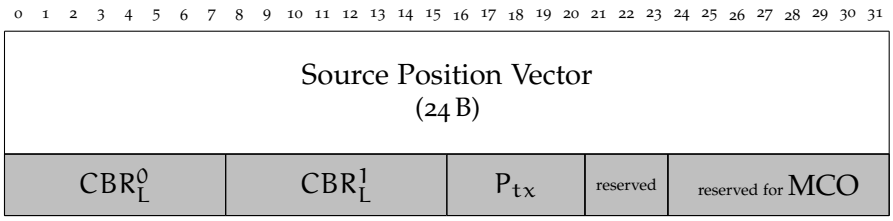


Figure 2.5: SHB extended header with redesignated DCC-MCO field (grey)

Congestion Control at Network Layer

TS 102 636-4-2 [195] specifies several aspects of GN when it is operating on ITS-G5 radios. With respect to congestion control, *TS 102 636-4-2* adds 'information sharing' through the DCC sub-entity at network layer (*DCC_NET*). This addition is twofold as it describes how information is disseminated as well as how received information is maintained. If 'information sharing' of CBR measurements is employed, input to DCC algorithms becomes the global CBR instead of local CBR as outlined in Section 2.1.4. Global CBR is defined as the maximum out of three values: local CBR, maximum CBR reported by one-hop neighbours, and maximum CBR reported by two-hop neighbours. To make a station aware of the maximum CBRs of its one- and two-hop neighbours, four reserved bytes of the SHB header are redesignated as 'DCC-MCO' field as shown in Fig. 2.5.

CBR_L^0 is the one byte representing the locally measured CBR. CBR_L^1 is the maximum of the aforementioned field as received by any neighbour during the last CBR measurement interval, *i.e.* the maximum CBR of one-hop neighbours. By analogy with the calculation of CBR_L^1, the maximum CBR of two-hop neighbours is then the maximum of received CBR_L^1 fields. This maximum is never transmitted but only used internally to determine the global CBR. P_{tx} encodes the transmission power of this SHB packet in 1 dBm steps, saturating at 31 dBm. The information conveyed by the DCC-MCO field is stored in an extended Location Table (*LocT*), *i.e.* its per-station Location Table Entries (*LocTEs*) comprise additional data elements. Related to P_{tx}, such a LocTE also stores the corresponding received signal strength. Since much of this standardisation effort has been clearly influenced by prior research work, the benefits of sharing CBR_L^0, CBR_L^1 and P_{tx} are discussed in relation to these prior works in Section 2.4.3.

2.2 DISSEMINATION OF V2X MESSAGES

Under harsh conditions, the transmission range of Vehicle-to-Anything (*V2X*) radios may become significantly shorter than the distance to relevant recipients of a V2X message. In Europe, IVC relies on collaborative dissemination by repeating these messages and thus forwarding them beyond a single

station's transmission range. GN offers several forwarding algorithms which control how messages are going to be repeated. Characteristic channel usage by particular algorithms also affects channel congestion in the end. This section analyses these algorithms with an emphasis on those favoured by standardisation. While the first part (Section 2.2.1) deals with the dissemination of messages within their addressed destination area, the follow-up part (Section 2.2.2) looks at keeping messages available over time. The distinction between both is thus spatial versus temporal dissemination of messages.

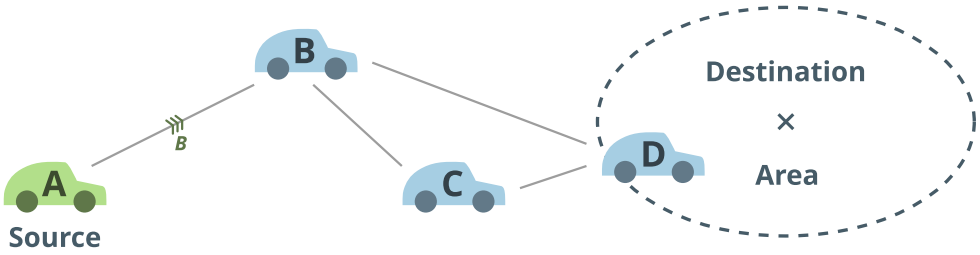
2.2.1 Packet Forwarding

When GN receives a multi-hop packet, its first decision concerns the routing mode. If the receiving station is within the packet's destination area, the *area forwarding* mode is chosen. Is the station located outside the destination area, it opts for the *non-area forwarding* mode. A router will discard the packet instead of proceeding with *non-area forwarding*, if the packet has been received from a station located within the destination area, though. This discarding avoids switching between modes when routing a packet through the network: If the originating router is already within its packet's destination area, *non-area forwarding* will not be employed at all. The destination area of a GBC or GeoAnycast (*GAC*) packets is encoded in its GN *Extended* header as geographical area shaped as circle, ellipse or rectangle. Equations to determine if a geographic position is located within, on the border, or outside such a destination shape are given by *EN 302 931* [60].

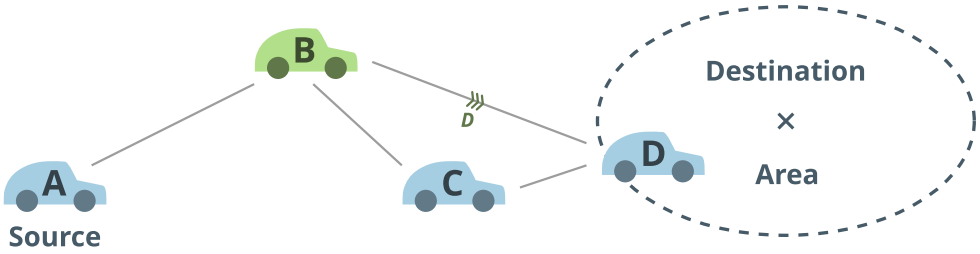
Non-Area Forwarding

Purpose of *non-area forwarding* is to bring the message closer to the destination area. Since the destination area does not address any of the forwarding stations located outside the destination area, the message will not be passed up by GN and is thus invisible to upper layers. GN offers two fundamentally different options for *non-area forwarding*. One is called Greedy Forwarding (*GF*), which is also the current default option; the other is called Contention-Based Forwarding (*CBF*) for *non-area forwarding*.

With GF the current router decides who will be the *next hop*, hence this forwarding is sender-based. The router looks up in its LocT, which directly neighbouring station is closest to the destination area. GF is thus one of the rare occasions where unicast addresses are employed as destination addresses at the link layer in an ITS-G5 VANET. The mechanics of GF are visualised by Fig. 2.6. If the current router itself is the local optimum, *i.e.* no closer neighbour is available or known, GF falls back to a plain broadcast or will buffer it for later transmission if SCF is enabled for this packet.



(a) Step 1: Source knows only one suitable next hop



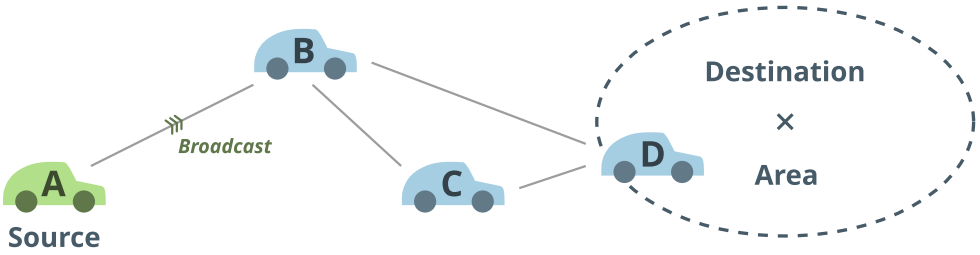
(b) Step 2: Among two candidates, the one with most progress is selected

Figure 2.6: Non-Area Forwarding with Greedy Forwarding

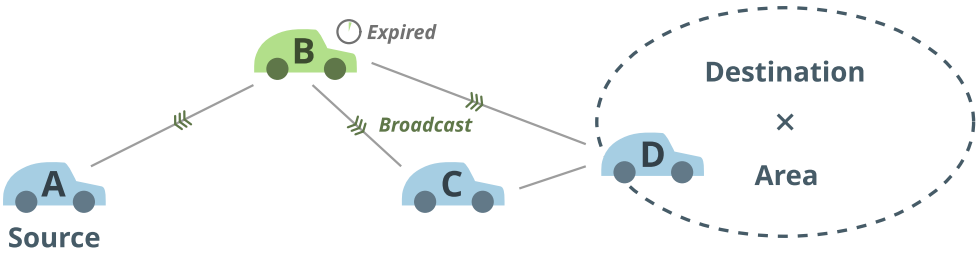
Management of the LocT and marking LocTEs as direct neighbours correctly is a prerequisite for GF. As soon as a packet is received from a previously unknown GN station, a LocTE is created for this station by the receiving router. Such entries are 'soft-state', *i.e.* they will be dropped if they have not been updated for more than 20 s. For every reception of a GN packet, the LocTE corresponding to the packet's originator is updated. To put it in another way, LocTEs of merely forwarding stations are not updated. Only if a packet has been received directly from its originator with certainty, this originator is marked as a direct neighbour in the LocT. Consequently, marking direct neighbours relies on the periodical reception of SHB packets such as CAMs. If a station does not transmit any SHBs for more than 3 s to 3.75 s, a GN beacon will be unsolicitedly sent by the GN layer. Since vehicles usually run a CA service, however, such GN beacons are usually not observed in the VANET.

Relying on unicast packets imposes the risk that the selected next hop is no longer reachable. LocTEs may contain outdated neighbour states because stations have moved apart beyond their radio's transmission range. GF does not specify any recovery procedures, *i.e.* a single failed unicast transmission prevents the GN packet from reaching its destination area at all.

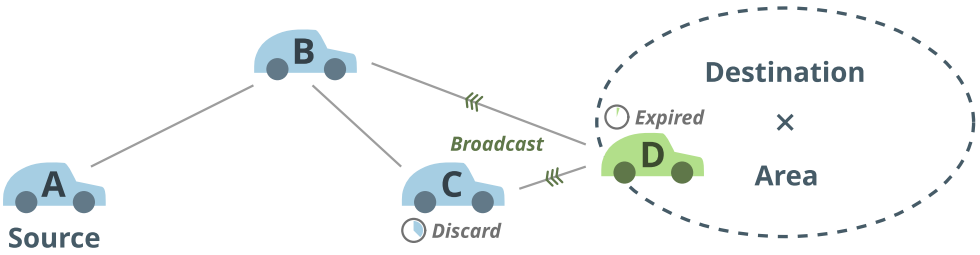
The CBF alternative avoids such failures and shifts the routing decisions from senders to receivers. When CBF is employed, the destination MAC address is always set to the broadcast address. Consequently, all physical receivers pass the to-be-forwarded packet up to their network layer where forwarding decision takes place. CBF does not immediately forward received multi-hop packets but puts the packet in a dedicated packet buffer, where



(a) Step 1: Source immediately broadcasts its packet



(b) Step 2: Though B is the only possible forwarder, it has to wait for timer expiry



(c) Step 3: Forwarding by D causes C to discard the packet

Figure 2.7: Non-Area Forwarding with Contention-Based Forwarding

each enqueued packet is associated with a timer. On the expiry of such a timer, which is set between 1 ms to 100 ms, the corresponding packet is re-broadcasted. However, the packet is immediately discarded when the same packet is received again by another forwarder. The timer duration is scaled inversely proportional to the progress of the packet towards its destination area compared to the previous sender: The larger this distance, the shorter the chosen contention period from the interval 1 ms to 100 ms. Those receivers with negative progress, *i.e.* which are farther away from the destination area, will not buffer the packet at all but discard it immediately. Receivers close to the previous sender with a little progress enqueue the packet with a rather long timer duration. If the progress is 1000 m or more, the shortest possible timer duration is used. The principles of CBF are also depicted in Fig. 2.7.

The desired effect of CBF is that those stations promising the largest steps towards the destination will quickly forward the packet by re-broadcasting it. Those retransmissions will be heard by less promising stations with longer timer durations, which consequently stop their scheduled forwarding intent.

However, CBF is also not invulnerable to local optima situations, and each forwarding step adds latency on principle. Both forwarding schemes make use of LocTs, but outdated entries only affect timer durations of CBF while GF may take entirely false decisions. If no suitable entry is found, CBF simply falls back to the maximum timer duration. In contrast to GF, a wrong decision by a single station cannot defeat forwarding entirely.

Area Forwarding

ETSI specifies three area forwarding algorithms for spreading a message within its destination area. The most basic variant simply re-broadcasts a GBC packet on reception immediately. Each forwarder tracks via Duplicate Packet Lists (*DPLs*) which packet, identified by its sequence number and source address, has been forwarded already. At most, each station within the addressed destination area will forward the packet once. Especially when the density of stations increases, the vast number of retransmissions in a short time frame comes with a high risk for packet collisions. Scalability of this simple forwarding algorithm is thus somewhat limited, while it features low processing costs and dissemination latencies.

ETSI suggests a CBF variant as default for area forwarding, which is very similar to its non-area forwarding counterpart. The essential difference in both CBF variants is the calculation of the respective timer duration. Instead of scaling the timer duration inversely by the 'forwarding progress', the distance between the previous sender and the current station is adduced. Consequently, distant stations tend to re-broadcast a message earlier while near stations discard the buffered packet according to CBF's overhearing. Since nearby station would only marginally help to disseminate the information all over the destination area, CBF employs these stations in packet forwarding only with comparatively long timer durations.

Under the designation 'advanced', ETSI amalgamates several enhancements over plain CBF. These enhancements do not depend on each other and could thus be employed independently. Indeed, the GN implementation in Vanetza (see Section 3.2.4) allows a more fine-grained configuration to facilitate the isolated evaluation of each enhancement introduced by 'advanced' CBF.

First, when a station within the destination area receives a packet specifically, *i.e.* it has been selected by the previous station's GF, this packet enters two routing paths: Continued GF towards the centre of the destination area with low latency, plus CBF to increase coverage. Especially destination areas of large diameters requiring multiple hops from its border to the centre can benefit from the faster GF path. However, this also doubles the channel load induced by forwarding a particular packet. From the viewpoint of congestion, no difference exists between unicast and broadcast link-layer frames.

Combining GF and CBF has probably been adopted by ETSI from Torrent Moreno [46]. In his thesis, he also motivates the reduced delay to select specific fast forwarders while keeping CBF as the fallback if reception by one of those forwarders fails. However, on the link layer, all transmissions are still broadcasted with his 'EMDV' forwarding protocol to avoid acknowledgement packets. Current GN header design does not allow to realise this protocol behaviour, though. Since each forwarding station changes the selected forwarder, the only viable option would be to add the GN address of the selected station to the *Basic* header. Otherwise, signature verification of these packets would break.

The second enhancement included in 'advanced' forwarding aims at reducing retransmissions by stations close together. Mariyasagayam, Osafune and Lenardi invented a mechanism coined as 'sectorial backfire' [45, 91], which has been incorporated in GN standardisation. As a prerequisite for this mechanism, a station R has to remember the location of the sender S from which it has received the packet. When this packet is received again during buffering from another forwarder F, the contention is stopped if the buffering station is within the same circular sector centred at the memorised sender position. Geometrically speaking, the angle between F and R at S has to fulfil $\angle FSR <= \Phi/2$ where ϕ is the sector's opening angle. On top of that, $\overline{SR} < \overline{SF}$ applies, *i.e.* R is closer to S than F. Ideally, the sector's opening angle is adapted to the current node density: A wider angle at higher densities reduces the impact of forwarding on channel congestion. However, no normative rules are given by ETSI standardisation how this adaptation shall be made. GN's table of protocol constants specifies $\phi = 30°$ as default.

Torrent Moreno [46] incorporates 'forwarding areas' in CBF, which are conceptionally similar to the backfire sectors by Mariyasagayam, Osafune and Lenardi. Where ETSI only considers forwarders up to the predefined, theoretical maximum communication range d_{cr} for one-hop transmissions, *i.e.* $\overline{SF} < d_{cr}$, Torrent Moreno argues that the range checks of forwarding areas should use a separate d_{fr} threshold, which is shorter than d_{cr}. Stations located in the margin between d_{fr} and d_{cr} are said to be susceptible to much interference. Hence, these stations are no forwarding candidates the algorithm should count on. Torrent Moreno evaluates the trade-off between reliability and overhead for $d_{fr} \in \{300\,m, 500\,m, 700\,m\}$, however, actual reception probability for a particular transmission range depends on many environmental influences.

The last addition to 'advanced' forwarding allows for a certain amount of redundant retransmissions. Instead of stopping contention immediately when a retransmission is heard, a counter will be incremented for the buffered packet. Only if this counter reaches a predefined threshold, the packet will be discarded by a station. In these cases, the station assumes a sufficient

number of nearby redundant transmissions of the respective packet, so that another retransmission would not noteworthy improve reliability further. Unfortunately, *EN 302 636-4-1* [193] gives no value for a reasonable redundancy threshold. Torrent Moreno [46] investigated reception probabilities for thresholds of 1, 2, and 3. Allowing only two duplicates, he has already observed probabilities well above 99 %.

Paulin and Rührup suggest to replace the hard threshold when to stop forwarding contention by a stochastic process. Whenever a redundant packet is received, the already buffering station tosses a coin if it shall cancel its forwarding procedure. The likelihood for abort grows exponentially with the packet counter N_P: $1 - \theta_P^N$. Paulin and Rührup use $\theta = 1/2$ in their simulations. They deduce that at most $\frac{1}{1-\theta} = 2$ duplicates per contention will be sent.

Kühlmorgen et al. study the performance of the 'advanced' routing algorithm and also the individual benefit of sectorial backfire and retransmission counters [109]. This study employs a counter threshold of 2. Their simulation results of highway scenarios indicate that the combination of these individual mechanisms does not necessarily perform best. In fact, sectorial backfire without retransmission counters outperforms 'advanced' forwarding in terms of end-to-end delay and coverage ratio, especially when facing high vehicle densities.

Stations can observe the progress of CBF forwarding by watching out for re-broadcasts by others. If a station does not re-broadcast a message, though, does not imply that it has not received the packet but just refrains from forwarding. Explicit reception notifications in VANET communication have been proposed, *e.g.* the 'PACK' protocol by Jiang et al. [40]. With 'PACK', outgoing packets piggyback a list containing identifiers of the recently received messages by the respective station. If an insufficient number of stations in relation to LocTEs acknowledges reception, senders may repeat the transmission. Unfortunately, such explicit notifications do not scale nicely with growing network densities. Message identifiers need to be sufficiently long to avoid number collisions and length of lists needs to grow if messages shall not remain unacknowledged due to list overflows. Though not specified at the moment, GN's packet forwarding may adopt the idea of coverage estimation by comparing the set of re-broadcasting stations with their positions stored in LocTEs.

The number of publications dealing with forwarding algorithms in VANETs and related disciplines is sheer overwhelming. In no case can this thesis treat packet forwarding exhaustively. With respect to practical relevance, packet forwarding is only considered as far as it is linked to the standardisation of GN. GN enhancements proposed in Section 4.1 specifically take their viability for future standard revisions into account.

2.2.2 Information Keep-alive Mechanisms

The primary task fo packet forwarding is to disseminate information as quickly and reliably among addressed stations. A metric to assess the performance of a particular forwarding algorithm could be the coverage ratio within a given time window, *e.g.* 90 % of all stations within the destination area in less than 0.5 s.

Keep-Alive Forwarding (*KAF*) fulfils a complementary purpose by keeping the information available in the network. After the plain packet forwarding is finished, newly arriving vehicles are unable to obtain information from still valid but no longer transmitted packets. With KAF these packets are repeated on the radio channel and thus actively kept alive.

In principle, KAF can be realised by services at Facilities layer but also at GN. The link layer is not suitable for this purpose because it has no notion about the validity of packet with respect to duration and destination. The Facilities' DEN service indeed outlines KAF operations for its messages [172]. The DEN message format contains not only the validity duration and relevance area, but also includes flags to terminate and cancel particular events identified by their 'action identifiers'. DEN services maintain the most recent update of a specific event in their local message table. As long as the message stored in this table is valid, *i.e.* has not been cancelled nor expired, it may get repeated if the source DEN service has specified a transmission interval. The DEN payload of these repetitions is the unmodified message as it has been received originally. However, each DEN repetition is a new transmission request for the GN entity, *i.e.* with a unique GBC sequence number. From the network perspective, such a repetition is just equal to an entirely new DENM.

GN itself also has built-in KAF features, including automated packet repetitions. The requesting service needs to specify the desired repetition duration and interval when it passes a message on to GN, to make use of this feature. Those repetitions apply only to the source station, though. Since aforementioned repetition parameters are not even encoded in GN headers, receiving stations cannot schedule repetitions properly, if they are intended at all. A further limitation of this rather simple GN mechanism is the lacking ability to abort pending repetitions, *e.g.* when service's logic determined that the information has become obsolete. Still, a more sophisticated KAF at GN would have compelling advantages: On the one hand, KAF is readily available for any service, not just DEN. Thus, such a design would also reduce the complexity of services. On the other hand, the cryptographic load could be reduced by avoiding to sign packets with equal payloads over and over. Even more, clever interweaving with packet forwarding may help to reduce the overall congestion.

In sparse networks, when no viable forwarders are available, SCF is another GN feature to improve packet dissemination over time. For example, GF may not find another station in the LocT closer to the destination. Without SCF, GF falls back to broadcast in the hope that a yet unknown station can still forward the packet. By enabling the SCF flag of a packet's TC, forwarding algorithms are allowed to buffer the packet up to the lifetime given in the *Basic* header. As soon as further stations come into transmission range, the packet forwarding operations are resumed, and the SCF packet buffer is flushed with packet lifetimes reduced accordingly. Consequently, a vehicle which may detect black ice while no other vehicle is within range will not broadcast such a warning futilely with enabled SCF.

An effort to increase dissemination rate of SHB safety packets is outlined by Jiang et al. [40]. Translated to ITS-G5 and CAMs their 'ECHO' proposal could be realised as follows: They assume that CAMs are sufficiently short (without overhead by security and other headers) to convey a prior received CAM in the same packet. With a slightly modified CAM format, the CA service may forward a previously received CAM by adding it as optional data container to its own CAM. Similar to CBF on the network layer, the CA service refrains from including CAMs which have already been repeated by others. CAM payloads are thus transmitted twice on average, according to Jiang et al. However, they give no numbers about the expected improvement of CAM reception rates nor deterioration of CAM information age. Considering the frequent generation of CAMs, it remains unclear if 'ECHO's' overhead is justified.

With respect to KAF, repeating mere payloads by piggybacking instead of repeating full packets is a way to reduce the average overhead to keep information alive. However, the aforementioned 'ECHO' protocol is designed for rather short-lived message types. In principle, a CAM may not only piggyback another CAM but arbitrary payloads as long as the total packet length does not exceed the Maximum Transmission Unit (*MTU*) of 802.11 frames. CA may then select the payload for piggybacking which is still valid but has not been repeated by anyone else for the longest period. Apart from the limitation in payload length with such KAF at the application layer, moving generic KAF out of the network layer can be considered bad design.

All in all, KAF mechanisms specified in GN are not very sophisticated. Flushing SCF buffers all at once when a neighbour arrives may easily lead to peaks in channel access. Of course, sparse network conditions where SCF is active at all do not suffer from channel congestion and thus can compensate such peaks. However, arriving neighbours may be part of a cluster, and thus further packet forwarding by this cluster could exacerbate dense packet transmissions.

2.3 QUALITY OF SERVICE

The term QoS, as used in this thesis, is the performance level provided by the network for a particular application. Similar to Xiao [51], this thesis distinguishes between QoS as a performance goal and QoS mechanisms to achieve this goal. However, Xiao opted not to distinguish between QoS and Quality of Experience (*QoE*). Latter is a networked application's performance from the perspective of end-users, according to Evans and Filsfils [44]. Speaking about ITS applications, *e.g.* a cooperative ADAS warning the driver of an imminent collision risk, the response time granted to the driver is a QoE metric. More time to react means a lower stress level for the driver and thus a higher satisfaction with this ADAS. Then again, too early warnings may be perceived as false positives and annoy the driver up to the point where the ADAS gets disabled entirely. A pleasant user experience thus does not only depend on the network's performance backing a cooperative ADAS but also on how it is integrated with the particular vehicle, exposed to the driver and last but not least on the driver's preferences. Because of this, QoE is kept separated from QoS, which is then solely related to the network-oriented and technical aspects, enabling a satisfactory user experience. In the context of this thesis, the focus lies on QoS, which applies to a VANET as a whole. QoS is thus the contribution of a VANET to QoE as perceived by users.

QoS requirements in a VANET derive from the employed ITS applications. Meeting QoS requirements is ultimately a problem of managing congestion, *i.e.* when network capacities exceed the load created by applications using this network [44]. In contrast to Internet applications, however, network load is not directly caused by users but by embedded components, *i.e.* the Facilities layer in the ITS-G5 architecture. Of course, a driver may provoke packet transmissions by causing accidents deliberately, for example, yet wireless communication by vehicle features is mostly hidden from the driver. To put it another way, instead of the driver pressing a 'transmit warning now' button, the vehicle is expected to issue such a message autonomously.

In the following, QoS and its measures found in ubiquitous IP networks are highlighted first. QoS metrics employed for IP can be used to assess VANET performance as well. Particular QoS challenges found in wireless (IP) networks and the importance of queue management are presented subsequently. Before finishing this section with a discussion on the implications of these related domains on IVC, the QoS requirements of V2X services as found in the literature are summarised.

2.3.1 QoS Metrics and Mechanisms in IP Networks

Networks based on the Internet Protocol (*IP*) are omnipresent, and thus it is worth to take a look at QoS offered by them. While the end-user experience itself is hard to measure, four metrics are usually employed in IP communications, which relate to the perceived network performance [51]:

END-TO-END DELAY or latency is the time span between the source application generating a packet and the destination application receiving this packet. This delay is determined by the physical propagation speed of the signal, packet processing delays in network devices, and the speed of the communication technology.

JITTER relates to end-to-end delay as it describes the variance in delay. Users tend to prefer low jitter in interactive applications since it eases prediction when a reaction can be expected.

PACKET LOSS occurs when a packet is either dropped at some point or bit errors prevent the successful decoding of a packet. Some transport protocols such as Transmission Control Protocol (*TCP*) can mitigate packet loss by retransmissions at the cost of increased latency. Applications that can deal with lost packets on their own can avoid this delay penalty by using User Datagram Protocol (*UDP*) as their transport protocol.

THROUGHPUT is the amount of transferred data per time and a result of the delay, jitter and packet loss ratio. Still, throughput is commonly used as users can perceive it when downloading files, for example. A TCP connection over a network with low delay and packet loss can serve data at a high rate.

QoS Mechanisms

Originally, all IP traffic has been processed equally according to the 'best-effort' principle. Real-time requirements for demanding, interactive IP applications could not be guaranteed, though. Treating some packets with preference over others introduced QoS in IP networks. Two principal directions of QoS measures exist for the Internet:

INTEGRATED SERVICES (*INTSERV*) is a deterministic QoS model offering end-to-end guarantees per flow. Such a flow is often identified by a five-tuple of IP destination and source address, the transport protocol identifier, as well as this protocol's destination and source port. An overview of IntServ's mechanisms is given by RFC 1633 [21]: Endpoints claim network resources by announcing the requirements of their injected traffic in terms of token bucket

parameters. These requests are handled by a two-phase reservation protocol, involving all routers between the endpoints of the particular flow. Besides resource reservation, IntServ also features admission control, *i.e.* flows can be denied to assure sufficient bandwidth to meet service levels of already granted flows.

Differentiated Services (*DiffServ*) is a QoS model which relies on traffic prioritisation. Its architecture is outlined in RFC 2475 [23]. Instead of fine-grained, per-flow QoS requirements, each packet is assigned to one traffic class, a DiffServ code point. Routers apply their preconfigured per-hop behaviour to packets according to their traffic class then. Such a per-hop behaviour is implemented by packet scheduling and queuing mechanisms. The implementation details are left to the network providers, *i.e.* networks may handle the same traffic class differently. Due to the lack of admission control, DiffServ can only guarantee service level boundaries if the peak load never exceeds the network's bandwidth. [44]

According to Xiao, IntServ's overhead of managing reservations causes scalability issues, and thus it has never been deployed widely and been virtually abandoned in favour of DiffServ. Reasonable over-provisioning of bandwidth enables the Internet to fulfil most QoS demands, though.[51, Chapter 5]

Beyond classification of traffic into flows, further accompanying QoS mechanisms are scheduling, policing and shaping. Simply put, scheduling is about determining the time point at which a packet is processed. Applying a policy on network traffic is about dropping or marking packets violating the enacted policy, *e.g.* a data rate limit. Traffic shaping is similar to policing in terms of imposing limits on network traffic. However, packets are not dropped immediately but buffered and processed later when resources become available again. Hence, traffic shaping is always backed by a queue, whereas policing is not and thus never delays packets. [44]

2.3.2 Challenges in Wireless Networks

Compared to wired networks, interference is a severe issue in wireless networks. Other users of the shared transmission medium may cause this interference, or it may be due to unrelated radio waves, which cannot be shielded. Consequently, QoS needs to be addressed differently. While the data rate of wired connections is constant in most cases, stations in a wireless network can switch between modulation and coding schemes to adapt to external circumstances. As Jiang and Liew conclude [37], this rate changing requires a different interpretation of fairness. They proved the equivalence of proportional fairness of bandwidth usage and max-min fairness with respect

to airtime usage. While the throughput of low-rate stations declines with airtime fairness, the overall throughput of stations connected to an access point increases. Plain max-min fairness concerning throughput is reached if increasing a station's bandwidth allocation is only possible by simultaneously decreasing another station's already smaller allocation. Proportional fairness strikes a balance between all stations achieving equal throughput and maximal total throughput. Optimising towards the maximum total throughput may lead to starvation of slow stations, though.

Similarly, Høiland-Jørgensen et al. [131] describe this phenomenon as 'Wi-Fi anomaly', when low-rate stations allocate a disproportional amount of transmission time in a WLAN. Høiland-Jørgensen et al. address the anomaly by a novel scheduler for access points, which replaces throughput fairness by airtime fairness. Though they were not the first identifying and tackling this issue, their approach actually made it into the Linux kernel and thus has found wide-spread usage. Remarkably, they have discovered voice-over-IP traffic sent with low priority (AC_BE) and proper scheduling plus queue management in place, outperforming equivalent high priority traffic (AC_VO) in terms of latency. By the use of contention graphs, Jiang and Liew also show that (distributed) airtime fairness in a wireless ad hoc network approximates proportional fairness (global) closely. Vertices in a contention graph denote MAC links and edges are possible collisions if the associated links are transmitting concurrently. However, equally important data traffic at a constant rate is assumed in their reasoning. Enforcing equal airtime usage among stations is thus not necessarily optimal with a mix of packet priorities as found in a VANET. Still, the fair long-term split of channel airtime among contending ITS stations is a reasonable goal.

Initially, no QoS mechanism was included in IEEE 802.11. Those were introduced later by IEEE 802.11e with EDCA, which allows differentiating traffic at the link layer. EDCA is part of the Hybrid Coordination Function (*HCF*), a QoS enabled channel access coordination function. In infrastructure mode with an access point, HCF enables this access point to take the role of a hybrid coordinator coordinating channel access centrally. However, EDCA remains the only specified, distributed QoS channel access algorithm. [121]

Carlson et al. [39] propose with DARE a distributed end-to-end reservation protocol for IEEE 802.11 networks to support hard QoS requirements. In contrast to the reservation protocol of IntServ, they take possible interference among nearby stations into account: All stations in the vicinity to the reserved path for real-time traffic has to keep silent at reserved times, *i.e.* not only capacity by relaying stations is reserved but also space. Obviously, all stations need to comply with restrictions implied by a protocol such as DARE to make this work. While it may be feasible to ensure such a unified deployment in a VANET's dedicated frequency band, those reservations require end-to-end

traffic in the first place, which is usually not found in VANETs. Maintaining reservations in view of a rapidly changing network topology is another weak point of such reservation protocols.

INSIGNIA [27], a QoS mechanism for Mobile Ad Hoc Networks (*MANETs*), considers the mobility of network nodes explicitly. It does not rely on additional control packets, *i.e.* out-of-band signalling, but sets options in IP data packets to achieve in-band signalling. Lee et al. claim that INSIGNIA can recover QoS reservations within two consecutive IP packets if a network route is still available. Applications can request a 'base' (min) and 'enhanced' (max) bandwidth from INSIGNIA. An adaptation policy may depreciate packets belonging to the 'enhanced' bandwidth when network conditions change. Only the 'base' bandwidth requirements remain in force then as available bandwidth shrinks.

Mohapatra, Li and Gui present an overview of how QoS is addressed in MANETs [32]. In summary, most approaches – including INSIGNIA – operate with flows from one source to one destination, which is suitable for the majority of IP traffic but not IVC. QoS mechanisms requiring changes at the MAC layer, such as 'Multihop Access Collision Avoidance with Piggyback Reservation' (MACA/PR) [22] or 'Black Burst Contention Scheme' [26], are not further investigated in this thesis as those would require changes at the Network Interface Cards (*NICs*). At this stage of ITS-G5 deployment, any changes breaking backward compatibility are unlikely to be accepted, especially at hardware components. Hence, this thesis focuses on tweaking the software stack to provide reasonable QoS.

2.3.3 The Bufferbloat Phenomenon

Bufferbloat, a phenomenon observed when large buffers fill up and will not drain anymore at bottlenecks of the network, impairs delay considerably [62]. Queue management can mitigate this problem by keeping queues short, which reduces queueing delays at the cost of increased packet drops. Packet drops are often not a big issue in Internet traffic, though, because TCP's congestion control actually relies on packet loss to detect congestion. Applications using UDP often have built-in mechanisms to deal with packet loss. In Internet communications, Active Queue Management (*AQM*) is the discipline of maintaining the length of queues. AQM drops packets strategically even before a queue fills up [101].

The Linux kernel has a feature called 'Byte Queue Limits' [114], which controls the number of enqueued bytes in front of the NIC, *i.e.* in the driver queue. This mechanism allows enqueueing only a dynamically determined upper limit of bytes. When this limit is reached, the network layer needs to decide either to drop the packet or to buffer it on its own. At periodic

intervals the limit is adjusted: It is increased if the hardware could have transmitted more, but the queue became empty or decreased if the NIC has been busy all the time and more packets arrived. The goal behind this strategy is to have a queue just large enough to prevent starvation but not prone to bufferbloat's latency issues. Byte Queue Limits do not replace AQM, though, as it applies only to a single queue being part of the NIC's driver. Both can be seen as a joint effort to reduce bufferbloat across the network communication stack.

RFC 7567 [101] issues recommendations and best practices regarding AQM targeting at the performance of the Internet. It criticises tail and head drop techniques as it leads to full, standing queues which cannot buffer bursts any longer. Even worse, high-rate flows may occupy the majority of buffer space and thus virtually lock out low-rate flows. Furthermore, RFC 7567 recommends combining AQM with per-flow scheduling for fairness, *e.g.* Fair Queuing or Deficit Round Robin. Ultimately, queue management and scheduling are complementary mechanisms: Queue management is controlling the length of queues by dropping packets when necessary. Scheduling needs to determine which packet, *i.e.* which queue's head, is to be sent next. In contrast to simple First-In-First-Out (*FIFO*) queues, AQM starts dropping packet early before buffers fill up. With this forced packet loss, which is a feedback type indicating congestion in TCP connections, congestion control can then kick in earlier and regulate packet rates before overload occurs. AQM is expected to respond to measured congestion but not application profiles, *i.e.* not to base its drop policy on packet rates or sizes. Manual tuning of parameters has been a known problem of early AQM schemes such as 'Random Early Detection' (RED) [19]. Modern AQM schemes auto-tune themselves to varying conditions and thus are not overly optimised for one particular case.

CoDel, short for Controlled Delay, is such a modern AQM scheme [77, 152]. While RED just reacts upon current queue lengths, CoDel looks at the delay of enqueued packets. It is independent of round-trip times and link rates as its input is the local sojourn time of packets, *i.e.* the period from entering until leaving the queue. CoDel strives to eliminate any standing queues, only mitigation of bursts and link utilisation are considered valid buffering reasons. Good link utilisation is achieved when the NIC does not idle, *i.e.* the next packet is immediately available after a transmission. Hence, CoDel will never drop any packets if the queue does not contain at least as many bytes as the NIC's MTU. Drop state is entered if the minimum sojourn time of all packets dequeued in the last interval is above the target threshold. Up to 5 ms are considered as acceptable sojourn times. When entering the drop state, the current packet exceeding the target sojourn time is dropped immediately. In drop state, either packet drops are scheduled stochastically, or the state is left

if the sojourn time falls below the target value. With recurring drops while in drop state, further drops are scheduled more aggressively in shorter time spans to reach the link's actual rate. Oscillations regarding drop state are avoided by only re-entering it when it has been off for a couple of intervals. Also, the previous drop rate is restored in this case. The flow queue scheduler fq_codel [145] incorporates multiple CoDel queues. This scheduler maps each flow based on its aforementioned five-tuple to one of the queues, ideally one flow per queue. Thus, flows are isolated, and bandwidth can be shared among them fairly via enhanced Deficit Round Robin [28]. Nichols and Jacobson point out that AQM as their CoDel cannot replace differentiated queuing, which is still necessary for real-time (low latency, low jitter) traffic. For this to achieve, they recommend delay-bound per-hop behaviour.

PIE, Proportional Integral Controller Enhanced, is another modern AQM control scheme [133]. A variant of PIE has been adopted as mandatory AQM in cable modems as part of DOCSIS 3.1 [134]. PIE also detects congestion by monitoring the queuing delay. However, in contrast to CoDel, it does not rely on per-packet timestamps necessarily. Instead, queuing delay can be estimated by dividing the number of enqueued bytes through the (past) departure rate. Tracking the delay is one of three crucial components in PIE. The other two components are the random dropping of arriving packets and the periodic updating of the drop probability. By default, PIE aims at a latency of 15 ms and also the update cycle is scheduled every 15 ms. Updates of the drop probability take the deviation from the target latency as well its trend into account. Packets get only dropped by PIE when it is active because of a detected congestion. As long as the currently observed latency is below half of the target latency and calculated drop probability is below 20 %, or less than two average-sized packets are enqueued, PIE remains inactive. Furthermore, short packet bursts, *i.e.* not exceeding 150 ms, are explicitly allowed if no congestion has been detected.

Abbasloo and Chao [157] propose an even simpler AQM, which is simply dropping packets at dequeue stage if its sojourn time exceeds a preset limit. They focus on real-time UDP traffic and conclude, that in their cellular environment the delay is shorter than with CoDel while TCP's throughput is reduced by 3 % to 40 %.

Høiland-Jørgensen, Hurtig and Brunstrom [108] investigate AQM on WLAN connections. WLAN operated in infrastructure mode, as found in office and residential environments, involves retransmissions and packet aggregation ordinarily. Former is a result of the higher risk for transmission error on a radio channel. Latter strives to increase link's throughput by reducing protocol overhead. Høiland-Jørgensen, Hurtig and Brunstrom observe that the necessary hardware queues to gather packets for aggregation adversely affect the latency and TCP congestion control when WLAN is the

bottleneck in a network. Ultimately, AQM schemes performing well in a wired setup fail to control latency in a wireless setup. Since no packet aggregation is employed in ITS-G5, those queues are not required in a VANET. Also, the traffic in a VANET is predominantly broadcasted, and thus end-to-end flows do not exist. Nevertheless, low latency is a key requirement for any safety applications, and thus avoiding excessive buffering should also be a goal in VANETs. Along with the findings by Høiland-Jørgensen et al. [131], this makes a strong case for putting effort into queue management and packet scheduling in VANETs.

2.3.4 QoS Requirements of ITS Services

Unfortunately, it is difficult to find any proven numbers regarding QoS requirements for ITS applications. The C2C-CC has published documents covering 'Triggering Conditions and Data Quality' of several use cases. However, even the safety-critical use cases about exchange of impact reduction containers and pre-crash information, do not explicitly state any requirements on network performance. Reading between the lines, latency in the order of 100 ms seems to be acceptable because the triggered DENM is repeated three times with a gap of 100 ms. This repetition is meant to improve the application's reliability in case of transmission errors. The reaction time of these pre-crash use cases, at least in terms of network latency, is quite relaxed because they are triggered when the computed Time To Collision (*TTC*) falls below 1.5 s [167, 168].

ETSI's ITS group has published documents about a few applications' requirements, namely Road Hazard Signalling (*RHS*) [87], Intersection Collision Risk Warning (*ICRW*) [142] and Longitudinal Collision Risk Warning (*LCRW*) [88]. These applications rely on the transmission and reception of CAMs and DENMs, *i.e.* build upon the Facilities layer's CA and DEN services. ICRW also consults messages from infrastructure (Signal Phase and Timing Extended Message (*SPATEM*), Map Topology Extended Message (*MAPEM*), and Infrastructure to Vehicle Information Message (*IVIM*)) to analyse the current risk. However, these applications do not generate any messages on their own but may only notify the DEN service when they detect a risk. All three applications demand an end-to-end latency of below 300 ms. It needs to be pointed out, that this latency is calculated from the moment of raw data acquisition in the transmitting vehicle to the time point of initiating an action at the receiving vehicle, either by displaying a warning or an automatic action. For obvious reasons, the available time for the actual communication, *i.e.* end-to-end latency from message generation to message decoding, is shorter but not explicitly given. For the transmitter, RHS requires a duration of less than 220 ms from data acquisition to channel access in 95 % of all cases during

critical traffic safety situations. An additional requirement is, that gathering all raw data shall not take more than 150 ms At most 70 ms remain then available for message generation, passing it through the communication stack including wrapping into a secured message, until reaching the transmitter's antenna. At the receiver side, LCRW and ICRW specifications demand the application processing and decision-making to finish in less than 80 ms. After subtraction of these data acquisition and processing bounds, 70 ms are left for all network communication in total. Still, this latency requirement is quite relaxed as is applies only to communication up to 300 m under Line-of-Sight (*LOS*) conditions with an uncongested radio channel. LCRW's specification even lowers the required range to 200 m under LOS if the channel is congested and traffic density is low in the own lane. For increased reliability, the RHS specification [87] demands, that the packet loss of consecutive packets under LOS conditions within the communication range (300 m by default) shall not exceed 5 %. All three applications further require to be able to disseminate updated DENMs at least every 100 ms in emergencies. Further requirements in these documents are not affecting the communication directly, but demand powerful hardware and software implementations, *e.g.* enough resources to handle at least 1000 received CAMs and DENMs per second.

A pre-standardisation study on Cooperative ACC [174] by ETSI assumes that the CA message rate needs to be increased to at least 10 Hz for this use case. Another considered variant even suggests message intervals as short as 30 ms. However, this variant is expected to depend on dual-receivers to spread the message load on at least two radio channels. In any case, all involved vehicles explicitly have to support DCC to control channel load. Except for the increased message rate, however, no QoS requirements are given.

A dedicated specification exists about the requirements of V2X services backed by Long-Term Evolution (*LTE*) mobile communications [135]. With respect to IVC, the general framework is quite similar to V2X services using ITS-G5: Message length varies between 50 B to 300 B for periodic broadcast messages and up to 1200 B for event-triggered messages. Furthermore, a maximum rate of 10 messages per second needs to be supported, equal to the maximum rate of the CA service. Otherwise, the requirements remain unassertive such as for the maximum latency of 100 ms between vehicles in general and 20 ms in case of emergency. The LTE network 'shall be capable' to achieve this, however, it remains unclear if this applies universally or if exempts exist, *e.g.* Non-Line-of-Sight (*NLOS*) conditions. Likewise, the demanded high reliability without retransmissions by the application layer and support for high density of vehicles are not quantified by any numbers. Regarding communication range, no distance is explicitly given, but it shall provide drivers with 'ample response time', mentioning 4 s as an example.

Assuming a speedy vehicle approaching a static obstacle with a speed of 250 km/h, this translates to a range of at least 278 m. This range is in good agreement with the 300 m required by RHS, ICRW, and LCRW.

Enhanced V2X services, enabled by 5G mobile communications, impose more rigorous functional requirements [136]. Four categories of such enhanced services are outlined in more detail in *TS 122 186*: Vehicle platooning, semi- and fully-automated driving, sharing of sensor data, and remote driving. In contrast to the LTE requirements in [135], high traffic density is explicitly mentioned with 3100 to 4300 vehicles per km² assuming the worst-case US Freeway scenario. Table 2.4 summarises the most interesting requirements for IVC using 5G technology, *i.e.* those use cases with the strictest requirement due to the highest degree of automation.

Table 2.4: Assortment of enhanced V2X services' requirements using 5G mobile communications [136]

Use case	Payload (application)	Message rate	Latency (max)	Reliability	Range (min)
Platooning	50 B–1200 B	30 Hz	10 ms	99.99 %	80 m
Collision avoidance	2000 B	100 Hz	10 ms	99.99 %	–
Emergency trajectory	2000 B	–	3 ms	99.999 %	500 m
Shared sensor information	1600 B	–	3 ms	99.999 %	200 m
		–	10 ms	99.9 %	500 m
		–	50 ms	99 %	1000 m
Remote driving	–	–	5 ms	99.999 %	–

Since these use cases are still subject to research and development, their requirements are not complete yet. Unfortunately, the rationale behind those requirements, such as the high update rate for collision avoidance, is also not given. The requirements on reliability as defined in *TS 122 186* [136], though, is problematic for safety use cases. Reliability is defined as the success probability that a packet of a certain length is transmitted within the given delay. High reliability may be achievable at large but not in rare, though critical situations, *e.g.* in a pre-crash situation with increased communication demand in general. It needs to be pointed out that the reliability requirement as employed by *TS 122 186* [136] differs from the reliability term used for the RHS application [87]. *TS 122 186* explicitly notes that requirements of each use case do not necessarily sum up because message types such as DENM cover multiple use cases and the Facilities layer may be able to optimise message handling.

Sharing of high-definition sensor data is the most demanding use case investigated by 5GAA, the 5G Automotive Association, which demands a latency below 10 ms in 99.9 % of all transmissions [155]. The associated payload of pre-processed sensor data is assumed to be of around 1000 B length. No hints regarding the packet rate are given, though.

2.3.5 Related Disciplines and Discussion

With the mainly informatory character of Day One applications in mind, it is understandable that no elaborate QoS requirements exist for those ITS applications. CAMs are generated frequently, so some lost CAM packets are not severe, especially as long as they trigger no critical vehicle actions. At the moment, applications using received CAMs are ADASs supporting the driver, but not taking over control of the vehicle. Most DENM use cases are quite delay-tolerant because several seconds to minutes may pass until the vehicle reaches the event location, *e.g.* the end of a traffic jam or a crash site. However, with increasing market penetration and more ITS applications entering the scene, channel capacities will become scarce. Application developers strive for more precise tracking of nearby objects and vehicles or tight control loops for vehicle platooning. Increased message rates and reliability are associated with these wishes. Anticipating these future demands proves difficult as the empty cells in Table 2.4 and [136] show.

The system architecture for 5G mobile communications provides a predefined QoS profile for ITS use cases [187]. Those profiles are named 5G QoS Identifiers in the 5G system. The ITS profile belongs to delay-critical services with a guaranteed bitrate, in particular, a latency below 30 ms is guaranteed for a maximum data burst volume of 1354 B every 2 s. Furthermore, the Packet Error Rate (*PER*) is promised to be less than 10^{-5}. The concept of guaranteed bitrates requires resource reservation and admission control, conceptionally similar to IntServ introduced earlier. As a consequence, the vehicles' device-to-device direct communication needs to be managed by a cellular network then.

Khelil and Soldani [97], affiliated with Huawei, argue that communication technologies using CSMA/CA do not support QoS at all. They promote device-to-device communication in a licensed paid frequency band instead with support by cellular base stations coordinating channel access among devices, *i.e.* vehicles in our context. Admittedly, hard QoS guarantees cannot be given by a CSMA/CA access technology as employed by ITS-G5. While IVC assisted by a cellular network promises to use frequency spectrum more efficiently, wireless communications remain sensitive to errors. As vehicles may lose connection to base stations at any time, Khelil and Soldani do not present any solution to how device-to-device communication can guarantee

QoS levels beyond those of IEEE 802.11 in these cases. When more sophisticated QoS mechanisms are harder to manage, the more straightforward solution has often won in the past, such as Ethernet over Token Ring or Asynchronous Transfer Mode (ATM) [51]. Only time will tell if the QoS model of 5G will become the only viable solution to enable those enhanced V2X services mentioned earlier. Furthermore, the postulated advantage of licensed over license-free bands remains fuzzy: Though ITS bands are license-free, they are not open to consumer devices but reserved for ITS use cases.

Partners of the 5G Automotive Association (*5GAA*) pursue 'predictive QoS' [156, 188], *i.e.* upcoming changes in Key Performance Indicators (*KPIs*) are reported to vehicles in advance. Applications can then proactively adapt to the future QoS provided by the mobile network. They have identified latency, Packet Delivery Ratio (*PDR*), data rate (uplink and downlink) and connectivity (yes or no) as primary KPIs, for which predictions shall be available. These predictions may be performed by vehicles locally by monitoring those KPIs or supplied by the mobile network, *e.g.* ahead of hand-over procedures between cells when the latency is likely to increase temporarily. Based on these predictions, a QoS flow will get notified which KPI is going to change in which time window and area. The predicted change is accompanied by an estimation accuracy of the assumed quality level. As noted in [188], however, current Release 16 ('5G Phase 2') by 3rd Generation Partnership Project (*3GPP*) cannot support predictive QoS yet. However, the concept itself would be a welcome feature for V2X communication in general. Applications are not condemned to react to QoS changes suddenly but could proactively plan ahead and thus avoid potentially unsafe conditions.

Scheduling transmissions in a wireless network perfectly is actually an NP-hard problem, regardless of the underlying technology. A perfect scheduling plan means no packet collisions and no unused channel resources in this case. Ma and Lloyd [24] reduce this scheduling problem to distance-2 graph colouring. As shown by Lloyd and Ramanathan [18], this graph colouring problem is NP-complete. They assume a graph where the vertices represent radios and the (undirected) edges links between them. The colour of the vertices represent particular transmission time points, *i.e.* transmissions of equally coloured stations collide if they are too close to each other. Since a station cannot transmit and receive at the same time, all immediately neighboured stations require distinct colours. Furthermore, a station cannot receive from multiple transmitters concurrently, so two-hop stations or distance-2 vertices need distinct colours as well. Hence, a collision-free schedule corresponds to a graph, where no vertices as close as distance-2 share the same colour. The optimal solution, *i.e.* using the minimum number of colours, is equivalent to a transmission schedule without any gaps in between. In light of this,

even device-to-device communication assisted by base stations will also only employ a sub-optimal, approximated solution of the described scheduling problem.

Admittedly, QoS provided by EDCA in terms of four priority queues (access categories) is only one piece of the puzzle regarding QoS in an 802.11-based VANET. It is an essential insight that airtime but not throughput is the resource which needs to be managed fairly in a VANET. This objective is substantially different from the notion of fairness usually found in wired networks. Consequently, throttling wireless stations by the number of packets or bytes is unrewarding. Instead, each station needs to track its actual airtime usage, *i.e.* take each transmission's data rate into account determined by the employed Modulation and Coding Scheme (*MCS*). Thus, monitoring a channel's CBR, which is the sum of airtime used by all contending stations, is a crucial component. However, CBR alone is a too abstract measure for assessing if airtime is distributed fairly because individual resource demand and composition of traffic's priorities vary. Local decisions about when, how often and which packets to transmit can still lead to overall airtime fairness if those decisions are taken in a globally agreed framework. DCC can thus be seen as such a framework establishing fair channel usage by a standard ruleset.

Exhaustive over-provisioning of channel capacities for QoS can be a viable approach in wired networks [51, p. 48], but this is no option in wireless networks where it would just waste radio spectrum. Flows are great to aggregate traffic belonging to specific end-to-end session connections, likely to be routed along the same path in a wired network. Adapting this concept to VANET communications is not straight forward because of mostly broadcasted packets. One may be tempted to group ITS traffic by its Application Identifier (*AID*); however, only a single AID exists for the wide range of DEN use cases with individual triggering conditions. However, the GN traffic class field may be a viable option to establish isolated flows, *i.e.* each traffic class getting a fair share of channel access. Even a tuple of GN traffic class and ITS AID could be employed to distinguish flows.

In some domains, QoS is approached quite uniquely. Instead of over-provisioning or reserving bandwidth, Trotta and Sciullo [154] take advantage of having control over the placement of aerial drones. They place their drones efficiently to maintain a stable video stream to a service vehicle. Despite efforts regarding autonomous driving, we cannot assume to gain a similar option in VANETs in the foreseeable future.

In the following, only QoS mechanisms applicable to VANETs are considered. Consequently, delays caused by in-vehicle data buses or insufficient performance of electronic control units are out of this thesis' scope. Since a VANET is not controlled by a network or mobile service provider, no ser-

vice level agreements guaranteeing QoS can be granted by an entity. Still, participants of the VANET are bound to common rules, either by mutual agreement or government regulation. The DCC cross-layer is a technical incarnation of common rules to achieve QoS in the ITS-G5 VANET. In the following Section 2.4, state-of-the-art of DCC will be introduced, and enhancements of DCC are proposed later in Section 4.2.

2.4 CONGESTION CONTROL

In the previous section about QoS, Evans and Filsfils [44] have been cited who claim that fulfilling QoS criteria is about controlling congestion. Without any bottlenecks, congestion control would be obsolete entirely. Unfortunately, a commonly shared wireless medium becomes easily such a bottleneck. The particular challenges found in wireless ad hoc networks are discussed first in this section. In the further course, this section addresses how these challenges are dealt with on various layers of ITS-G5.

2.4.1 Challenges in Wireless Networks

Communication over a radio medium is less reliable compared to wired networks because the physical impact from outside can hardly be controlled. Besides various forms of interference, the channel access scheme of the 802.11 MAC does not scale infinitely.

The backoff procedure of 802.11 tries to avoid packet collisions, but it cannot guarantee collision-free channel access. If backoff counters of two or more stations reach zero at the same time, a receiving station within sensing range experiences a packet collision because of concurrent transmissions. Smely et al. [111] show the relation between channel load, collisions and channel access delay assuming 200 stations, all within mutual transmission ranges. In Smely et al.'s setup, CBRs below 40 % pose no difficulties as the number of collisions is very low and channel access delay remains below 10 ms in the worst case. Further increasing stations' message rates leads to distinct growth in reception errors and latency. Notably, CBR does not grow beyond ~85 % and the peak reception rate is reached shortly before at ~82 % channel load. *TS 102 687* sets a target CBR of 68 % for the adaptive DCC approach or applies the most severe restrictions above 60 % channel load alternatively [143].

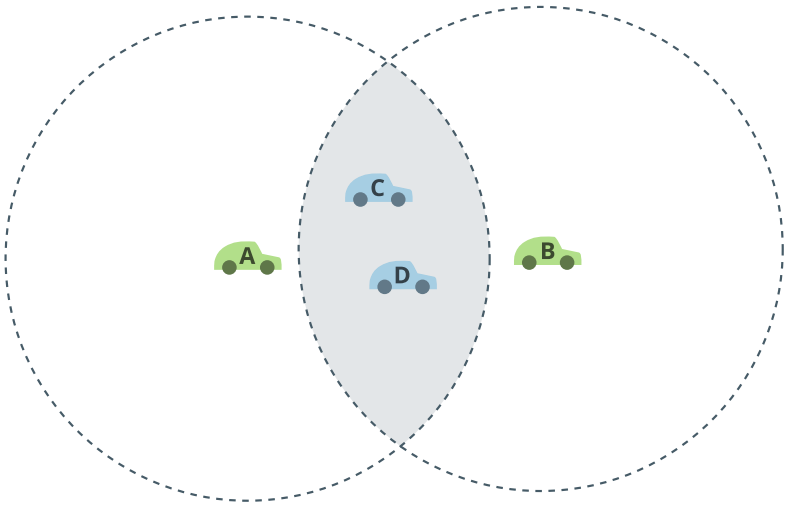


Figure 2.8: Schematic illustration of the hidden station problem

Subramanian et al. [79] seek for a target CBR, that strikes a balance between channel utilisation and a low amount of packet collisions. In their simulations based on ns-2, they observe that the total number of receptions levels off at about 65 % and even declines when approaching 80 % for all vehicle densities. Thus, they suggest a universal target CBR of 65 %.

A classic problem found in wireless communication based on Carrier Sense Multiple Access (*CSMA*) is the hidden station phenomenon[1], as shown in Fig. 2.8. This example assumes that stations A and B have pending transmissions. Both are outside of mutual sensing range, so the 'listen before talk' approach fails to detect ongoing transmissions. Receiving stations in the grey-shaded 'deaf zone', where transmission ranges of A and B overlap but neither is located inside, may be unable to decode messages when A and B transmit at the same time. Sjöberg, Uhlemann and Ström [67] have investigated the impact of concurrent transmissions in VANETs specifically. In their evaluation, a transmission is decodable by a receiver if the signal's power is 6 dB above the noise and interference level ($\mathrm{SNIR} \geqslant 6\,\mathrm{dB}$). Concurrent transmissions thus do not lead to a total loss necessarily. Packets in a VANET are usually addressed to all nearby stations as broadcast frames in terms of the MAC layer. While unicast frames can be acknowledged explicitly, the sending stations never gets any feedback about the reception for its broadcast frames. However, Sjöberg, Uhlemann and Ström conclude that concurrent transmissions by hidden stations do not devastatingly compromise packet reception probabilities. Though adjusting transmission power can help to achieve a desirable SNIR, they acknowledge that selecting an appropriate output power for a group of receivers is difficult.

1 Also called 'hidden terminal problem'

The phenomenon that at least one (strong) transmission out of several, concurrent transmissions can still be decoded, is called capture effect. How much stronger a signal needs to be for this effect, is expressed by the capture threshold as SNIR. Bazzi et al. [123] argue that hidden stations and the capture effect are crucial elements to be included by any 802.11p model.

Similarly, Torrent-Moreno et al. [209] investigate the probability of successful packet reception with both hidden stations and capture effect taken into account. They further point out that the employed radio propagation model has a significant impact on the packet reception probability. Packet level incoordination, their metric quantifying the impact of hidden stations, stands out more when channel attenuation models with a higher attenuation variance are employed. With strictly deterministic models, channels sensing and transmitting are equally affected as long as positions have not changed meanwhile. If the attenuation has a strong indeterministic component, however, sensing may be attenuated more than the follow-up transmission.

Cozzetti et al. [72] develop a radio propagation model for IVC on urban grid topologies with hidden terminals in mind. They point out that in urban environments, shadowing effects by building and other obstacles are much more pronounced compared to highway scenarios. With the increased influence of attenuation, transmissions are also more likely to collide because of hidden terminals. Hence, radio propagation models only considering the distance between stations but not actual obstacles, are deemed unrealistic. Though the effect of hidden stations is exacerbated on the one hand, the range of congestion is restricted on the other hand.

Torrent-Moreno, Jiang and Hartenstein [36] also observe a significant impact by hidden stations when using non-deterministic radio propagation models. This impact also finds expression in longer channel access times, *i.e.* a station's backoff procedure will be paused more often. Furthermore, they observe that reception rates decline rapidly in the last third of the transmission range. Attenuation of the signal is then already pronounced enough so that the SNIR threshold of the capture effect is reached less likely. [36]

In summary, it can be stated that many aspects interact with each other, which cannot be investigated separately. Channel access following the CSMA/CA pattern is affected by unwanted interference from hidden stations but also from the favourable capture effect. While these two are inherent to the PHY and MAC protocol, the attenuation characteristics of the wireless medium also play a significant role in this regard. *TR 103 257-1* gives an overview of channel models suitable for wireless communication in the 5.9 GHz band. This document outlines major reasons for varying attenuation typically employed in VANET scenarios. Probabilities for LOS paths and scatterers are considered in urban, rural, highway and tunnel environments [173]. The

topic of channel models will be revisited in Section 3.2.1, with focus on those models employed in this thesis specifically. The behaviour of the modelled radios, including CSMA/CA, is further detailed in Section 3.2.3.

Whereas Sjöberg, Uhlemann and Ström [67] suggest adapting transmission power to counteract hidden station's interference, Torrent-Moreno, Jiang and Hartenstein [36] propose packet repetitions and multi-hop retransmissions to achieve reliable communication. The former approach improves the reception probability measurable at the MAC layer. The latter's improvements are only measurable at the network layer and above. GN can compensate some lost frames at the link layer because packet repetitions are an integral part of its CBF packet forwarding scheme. Retransmission by a nearby station may be located outside the 'deaf zone' of (previously) hidden stations, or its position is favourable in terms of signal attenuation. It follows that isolated investigation of single layers is unrewarding as it does not capture the system's performance. ITS use cases, however, rely on ITS-G5 in its entirety.

Blazek et al. [161] focus on bare 802.11p radios in their work. They conclude that challenging radio channels with considerable interference demand for supporting control mechanisms by upper layers. They express their hope that DCC will ensure reception quality by adjusting transmit power, transmit rate and receiver sensitivity.

In the following sections, various mechanisms and ideas from literature are presented how to mitigate interference and collisions by keeping the congestion of the radio channel within bounds. Within the ITS-G5 architecture, those mechanisms belong to DCC, a cross-layer ranging from link layer up to the application layer. DCC, as found in standardisation, is covered by Section 2.1.4.

2.4.2 Mechanisms at Access Layer

Available mechanisms to mitigate channel congestion at the access layer comprise a wide range of radio parameters. Current standardisation in *TS 102 687* [143], however, names only TPC, TDC and TRC explicitly. Reactive and adaptive DCC, the two approaches outlined in *TS 102 687*, do neither modify transmission power nor data rate, though. Hence, DCC as per *TS 102 687* boils down to TRC only. Earlier versions of *TS 102 687* [61] described more adjustment screws but adapting more than message rates proofed to be complicated. Autolitano et al. [81] observe even sub-par performance when multiple DCC mechanism are combined as unfavourably as in *TS 102 687* [61].

The plethora of transmission parameters, adaptation strategies and possible combinations of them makes it infeasible to cover literature about these exhaustively in this thesis. Nevertheless, this section recapitulates some exceptional concepts so the reader will be able to conceive the overall picture.

Controlling Packet Rate

Changing the packet rate is the primary way to scale the shared, limited bandwidth among wireless network users. Controlling the rate of outgoing packets can be seen from two perspectives: For one thing from the application's perspective generating the message, for another thing from the link layer's perspective sequencing the channel access for those generated messages. The former perspective will be discussed in Section 2.4.4 thoroughly, which deals with DCC at the Facilities layer. Latter is DCC at the access layer, which is often nicknamed 'gatekeeper' as it may defer or drop packets arriving from upper layers if the station is going to exceed its channel resources.

REACTIVE TRC Autolitano et al. [81] investigate reactive DCC according to [61], *i.e.* backed by a finite state machine. In many situations with low to medium traffic, they find the DCC parameterisation picked by standardisation overly conservative. Notably, the most restrictive state is already entered at a channel load of merely 40 %. Even uncontrolled IEEE 802.11p performs better in those situations with respect to packet delivery ratios. Although TRC is found to improve the packet delivery ratio in high-density traffic scenarios, it comes at the cost of greatly expanded CAM update delays. Another shortcoming is the observed instability of the reactive approach, *i.e.* stations transition back and forth frequently instead of settling down at one state. With only three states, parameter changes from one state to another are too coarse to control channel load adequately. Similar results are obtained by Eckhoff, Sofra and German [85]. They also observe volatile state machines and too conservative settings when many vehicles are involved. As a result, considerably fewer CAMs are received with increasing penetration rate reducing the mutual awareness.

Increasing the number of *active* states can only partially improve the reliability of VANET communication in terms of range and end-to-end latency, as observed by Gómez and Mecklenbräuker [116]. When two vehicle clusters are approaching and later passing by each other on a highway, they still recognise stability issues of reactive DCC with five *active* sub-states. For a time window of several seconds, DCC fails to achieve an end-to-end latency below 100 ms for at least 90 % of packets. Nevertheless, Gómez and Mecklenbräuker value the better 'data novelty' – in terms of reduced average message age – achieved with TRC compared to plain 802.11p without any DCC.

In order to overcome the overly conservative state machine parameterisation by ETSI, Lyamin, Bellalta and Vinel [201] propose an analytical approach to tweak transmission rates allowed by each state. Assuming the number of neighbours is known, *e.g.* from GN's LocT, they provide equations to determine the message rate for a given message length, number of stations and targeted CBR. While Lyamin, Bellalta and Vinel can determine sound TRC parameters this way, uniform traffic by a single application is presumed to predict the resulting CBR. Hence, this approach still does not fit to adaptive multi-service systems, which transmit messages of varying length in non-uniform patterns.

ADAPTIVE TRC Kenney, Bansal and Rohrs present LIMERIC [63, 83], a linear message rate control algorithm. This algorithm has been later adopted by *TS 102 687* [143] as the adaptive approach for DCC.

The original aim of LIMERIC is to find a common message rate among stations in a VANET so that the resulting CBR is always close to a given target CBR. Kenney, Bansal and Rohrs [63] assume all stations to generate only periodic messages of equal size, *i.e.* about 3 kB length. Without requiring to know the total number of stations nor their current message rate, the authors take advantage of the observation that a linear relation between CBR and the summed message rates exists. Stations then adapt their message rate at regular update cycles: They increase their message rate if the measured CBR is below the target, and vice versa.

TS 102 687 [143] adapts LIMERIC and employs per-station channel occupancies δ instead of message rates as given by Eqs. (5) to (7). Input variables of these equations are $\mathrm{CBR}_{\mathrm{input}}$ and the channel occupancy as calculated in the previous update cycle, δ'. All other parameters are fixed. Channel occupancy is an abstraction from message rates as it does not prescribe a particular packet length or a strictly uniform distribution of message generation time points. Instead, stations can freely arrange when and how to consume the airtime associated with their channel occupancy limit between update cycles. ETSI specifies an update cycle interval of 200 ms, *i.e.* $\delta = 0.02$ allows a station to occupy the channel for $\delta * 200\,\mathrm{ms} = 4\,\mathrm{ms}$. If all n stations consume their 'airtime budget' entirely, the resulting CBR is $n * \delta$. Assuming fixed packet lengths and bitrates, the granted airtime is equivalent to the message rate.

Kenney, Bansal and Rohrs discuss in [63, 83] how α and β influence the convergence behaviour of LIMERIC and how these parameters should be selected. They proof convergence for parameter settings fulfilling $\alpha + K * \beta < 2$, where K denotes the number of stations. They recognise that 'it is not possible to choose static values of α and β for which stability is assured' [63] in any scenario, *i.e.* with a widely varying K. Instead of adjusting these

parameters dynamically, they add gain saturation to their algorithm, which limits the magnitude of change at each update and thus stabilises it. Gain saturation limits are employed in Eq. (6) as G^{-}_{max} when channel occupancy δ gets reduced and G^{+}_{max} for increments.

TS 102 687 [143] forces calculated δ into the $[\delta_{min}, \delta_{max}]$ interval by Eq. (5). The lower bound is meant to prevent starvation when channel capacity is split among too many competing stations. In such cases, a minimum duty cycle of $\delta_{min} = 0.06\,\%$ is guaranteed to every station even if this results in a larger channel load than targeted by CBR_{target}. With the parameters set according to [143], CBR_{target} will be exceeded when more than 1133 stations compete for channel resources. The upper bound δ_{max} enforces the maximum duty cycle given by [128].

Furthermore, *TS 102 687* [143] demands synchronised update cycles for the adaptive approach inspired by LIMERIC. Kenney, Bansal and Rohrs [63] introduce LIMERIC using synchronised updates and proof its convergence based on this assumption. However, they also explicitly investigate asynchronous updates and are very confident regarding LIMERIC's convergence behaviour. Convergence is even slightly more stable in their simulations, though they do not prove the convergence properties for this case. With LIMERIC acting a gatekeeper, the opening times need to be carefully spread over time if synchronous updates are employed to avoid artificial clustering of packet transmissions. Why ETSI decided to add complexity for handling gate opening times remains a mystery, especially as synchronous updates are harder to realise in the field.

$$CBR_{\Delta} = CBR_{target} - CBR_{input} \tag{4}$$

$$CBR_{target} := 0.68$$

$$\delta_{clamp}(\delta_i) = \min(\delta_{max}, \max(\delta_{min}, \delta_i)) \tag{5}$$

$$\delta_{max} := 0.03$$

$$\delta_{min} := 0.0006$$

$$\delta_{offset} = \begin{cases} \min(\beta * CBR_{\Delta}, G^{+}_{max}) & \text{if } CBR_{\Delta} > 0 \\ \max(\beta * CBR_{\Delta}, G^{-}_{max}) & \text{otherwise} \end{cases} \tag{6}$$

$$\alpha := 0.016$$

$$\beta := 0.0012$$

$$G^{-}_{min} := 0.0005$$

$$G^{+}_{max} := -0.000\,25$$

$$\delta = \delta_{clamp}((1-\alpha) * \delta' + \delta_{offset}) \tag{7}$$

While reactive DCC suffers from stability issues, stations converge with LIMERIC to a mutual rate independent of vehicle density in simulations conducted by Bansal et al. [95]. Consequently, CA benefits from smaller inter-packet intervals. The same group of authors also study mixed networks, *i.e.* where reactive DCC and LIMERIC are deployed next to each other [102]. According to their findings, both DCC variants can coexist without negative impact on each other. Quite the contrary, mixing those variants can mitigate artificially synchronised transmissions. It has to be noted, however, that the parameter settings of LIMERIC with $\alpha = 0.1$ and $\beta = 0.033$ do not match *TS 102 687* [143] and only CAMs are considered in both simulations.

Bansal and Kenney [82] also show that LIMERIC can be employed when stations shall gain different rates based on weights. A station, such as an emergency vehicle, may be given greater weight and thus permanently be able to transmit at a higher rate. Weight of a station i can be described by the ratio $\frac{\beta_i}{\alpha_i}$. The ratio of stations' weights equals the ratio of their message rates then. Consequently, Eq. (8) applies this to channel occupancy limits.

$$\delta_1 : \delta_2 : \ldots : \delta_k = \frac{\beta_1}{\alpha_1} : \frac{\beta_2}{\alpha_2} : \ldots : \frac{\beta_k}{\alpha_k} \tag{8}$$

Such an enhanced LIMERIC opens up new possibilities as a distributed algorithm for resource allocation according to weighted max-min fairness. How those weights should be determined has not been studied so far to the best of author's knowledge. Sudden short-term violations of LIMERIC constraints may be silently ignored for simplicity, *i.e.* without adjusting weights. Those violations should occur rarely but may be acceptable for use cases such as an emergency DENM at an imminent crash.

Soto et al. [183] investigate the adaptive DCC of *TS 102 687* [143] specifically. They acknowledge that the lower $\alpha = 0.016$ chosen by ETSI helps to achieve a high channel utilisation, *i.e.* measured CBR approaches CBR_{target} closer. An advantage of a higher $\alpha = 0.1$, as used by the original LIMERIC paper [63], is the faster convergence. Soto et al. found the slow convergence with $\alpha = 0.016$ problematical when groups having different levels of δ approach each other. When stations of the merged group only slowly converge to a common δ, channel resources are unfairly distributed among them during this period. Soto et al. propose a dual-α enhancement by adding Eq. (9) as the final step of the LIMERIC algorithm after Eq. (7). Equation (9) refers to the result of Eq. (7) as $\tilde{\delta}$. If the allowed channel occupancy is decreasing significantly, *i.e.* beyond the heuristically determined threshold $\hat{\delta}_\Delta$, the new δ of this update cycle is calculated with a higher $\hat{\alpha} = 0.1$. Consequently, in a scenario with merging vehicle groups, those stations with a higher initial δ will reduce it more eagerly. Otherwise, if channel occupancy is close to convergence or increasing, the original α is employed by passing through $\tilde{\delta}$. Amador et al. [190] evaluated the dual-α mechanism more thoroughly, where they compare

achieved packet delivery ratios and inter-packet intervals when using dual-α and different sets of α and β combinations found in related work. They recognise that ETSI's original parameterisation performs better across a wide range of vehicle densities than other parameter sets. Dual-α shines when vehicle densities change, where the static parameterisation by ETSI unveils a weakness due to its slow convergence. Amador et al. admit that despite the empirically shown advantages of their dual-α approach, one loses the theoretically proven properties of the original LIMERIC. Considering the publication on weighted LIMERIC [82], however, changing α does not affect the stability of the LIMERIC algorithm but lets it converge at another δ.

$$\delta = \begin{cases} \delta_{clamp}((1-\hat{\alpha}) * \delta' + \delta_{offset}) & \text{if } \delta' - \tilde{\delta} > \hat{\delta}_{\Delta} \\ \tilde{\delta} & \text{otherwise} \end{cases} \tag{9}$$

$$\hat{\alpha} := 0.1$$

$$\hat{\delta}_{\Delta} := 1 \times 10^{-5}$$

Though LIMERIC achieves to keep CBR close to the target only by TRC, it originally builds upon the assumption of uniform messages, i.e. of the same length and same rate. *TS 102 687* softens this limitation as it substitutes message rate by a more generic channel occupancy limit [105]. Unfortunately, this specification does not give any hint on how non-periodic messages should be incorporated into this concept. Most ITS-G5 stations even at *Day One* deployment will already generate DENMs occasionally, not to mention the forwarding of DENMs's generated by other stations. Limited suitability of adaptive DCC in its current state when additional background traffic is sent (for example due to packet forwarding) has also been recognised by Amador et al. [190]. Up to now, it remains an open question of how to spend available airtime (δ) for these use cases efficiently, *i.e.* without wasting airtime nor exceeding limits.

Sensitivity of Clear Channel Assessments

Previously, DCC allowed adjusting the sensitivity of Clear Channel Assessment (*CCA*), *i.e.* when a receiver determines a channel to be busy or idle. While this sensitivity threshold affects the channel access, it is independent of the CBR calculation. Power control at the transmitter and CCA sensitivity control at the receiver side are two sides of the same coin: Either a receiver may turn itself 'deafer' to overhear interference caused by others and thus gain channel access more comfortable, or a transmitter reduces its output power to reduce the interference range of its transmission. However, unfairness becomes a problem quickly as a station with higher transmission power

and lower CCA sensitivity may put another station to silence forcefully. Tielert [100] decided to keep CCA threshold fixed while adjusting transmission power levels and claims not having observed fairness issues.

Schmidt et al. [66] investigate the benefit of reduced packet collisions when adapting the CCA threshold in VANET communication thoroughly. They use a low threshold at low vehicle densities and thus enable more reliable communication over longer distances as stations will listen for weak, distant signals more carefully. With increasing vehicle density, this threshold is gradually increased to keep channel access delays low. However, only transmissions of CAMs are subject to changed thresholds. DENMs always employ the highest threshold, *i.e.* those transmissions tend to ignore weak third-party transmissions in favour of short channel access delays. The CCA threshold is essentially adapted by a CAM's sojourn time in the local transmission queue: Initially, each pending CAM transmission starts with a base threshold of −85 dBm. If the CAM is still enqueued after (50 ms, 25 ms, 12.5 ms), the threshold is increased by (6 dB, 12 dB, 18 dB). Improvements according to their custom 'awareness quality' metric are demonstrated by simulations for all low, medium and high vehicle densities, *i.e.* 76, 166 and 266 vehicles per km². Schmidt et al. claim that their CCA threshold adaptation is becoming part of *TS 102 687*. Recent versions of *TS 102 687* do not consider CCA threshold adaptations in general anymore, though.

Experiments with Artery (see Section 3.2) using a similarly high vehicle density (about 275 vehicles per km²) on the *Griddy* map add another perspective to this topic. Results printed in Table 2.5 confirm that increasing the CCA threshold effectively lowers the latency of a transmission. The downside of high CCA thresholds is the reduced range as stations accept more interference when transmitting. The reception numbers for the 5349 broadcast transmissions underpin this observation. Across all variations of (static) CCA thresholds, one can observe a correlation between excessively congested channels with CBRs of 80 % and more and growing latencies. As long as channel load remains below 60 %, though, latencies never exceed 8 ms.

Adjusting CCA thresholds enables stations to acquire channel access more easily when exposed to interference, however, it does not help to keep interference bounded in the first place. Considering these findings and the fact that the adaptation of CCA thresholds has been dropped by standardisation, this particular mechanism is not further studied in this thesis.

Power Control Strategies

Ideally, TPC should select a transmission power just high enough that the signal remains decodable by the intended receivers. However, simply selecting a high output power permanently introduces unnecessary channel

Table 2.5: Latencies, reception counts, and CBRs for CCA threshold experiments

CCA (dBm)	Tx power (dBm)	Latency quantiles 99 % (ms)	99.9 % (ms)	100 % (ms)	CBR quantiles 99 %	100 %	Receptions in total
-65	20	0.53	0.88	1.25	0.49	0.53	1 057 917
	30	1.12	2.02	2.92	0.80	0.83	1 578 943
-75	20	1.07	1.50	1.80	0.51	0.56	1 142 919
	30	2.16	3.54	4.14	0.84	0.88	1 740 350
-85	20	2.22	3.46	4.25	0.54	0.58	1 309 814
	30	12.47	22.77	28.89	0.88	0.90	1 882 206
-95	20	3.78	6.62	7.82	0.56	0.60	1 416 066
	30	21.93	36.21	44.71	0.86	0.89	1 783 196

load for distant stations. Signal attenuation is hard to predict accurately, though, due to a wide range of environmental influences. Calculation of the achievable range for a particular transmission power is thus more of a rough guess. Yet, TPC is effective against hidden station interference [67, 209].

Standardisation allows transmission powers as low as 3 dBm up to the channel's maximum, *e.g.* 33 dBm in the primary G5-CCH in Europe [128]. A special case of TPC applies nearby tolling stations, which operate in a sibling frequency band at 5.8 GHz and are already deployed in the field. Within so-called 'protection zones', transmission power of ITS-G5 communication is limited to 10 dBm. Protection zones can be loaded from databases, or ITS-G5 stations may receive a special protection zone container in CAMs generated by Road-side Units (*RSUs*) colocated at these tolling stations. Implementing this protection mechanism is mandatory. However, the impact is limited to well-defined zones whose radius cannot exceed 255 m. [105]

Subramanian et al. [79] investigate a reactive DCC strategy implementing TPC using a finite state machine with six states, as shown in Fig. 2.9. The employed state machine will transition up to a more restrictive state if the lowest CBR within last 5 s exceeds 65 %. Transitions down to a more relaxed state occur if the highest CBR within the last second has been below 55 %. They note that this approach enhances the packet delivery ratio in highly congested environments, *i.e.* 900 vehicles per kilometre on a six-lane highway in their setup. Packet delivery ratios raise from around 35 % using uncontrolled 802.11p up to 60 % with enabled TPC for vehicles separated by 50 m. However, they also note that vehicles' state machines do not converge to a common state despite their uniform distribution. Hence, the achievable range is not shared fairly among vehicles because of diverse power level restrictions. Subramanian et al. suspect that the failing convergence may be an inherent issue of DCC approaches employing state machines.

Similar to Subramanian et al. [79], Gómez and Mecklenbräuker [116] adapt the transmission power from 4 dBm to 33 dBm throughout eight states in total. The power levels have been selected according to expected sensing ranges, given their radio propagation model and a CCA threshold of −90 dBm. However, the employed model is entirely deterministic and only accounts for the distance between stations. Performance of their TPC with less predictable transmission ranges for a given power has not been studied.

Tielert et al. [93] claim that an optimal transmission power exists, which is minimising the inter-reception interval for a particular communication distance. In contrast to the transmission rate, this transmission power is independent of the vehicle density. Their combined TPC+TRC algorithm favours transmission power over transmission rate then: The algorithm determines the desired transmission power for a given target distance first. If the channel load is below the predefined maximum, the packet rate is gradually increased. Vice versa, the algorithm counteracts exuberant channel load first by reducing the rate down to a minimum preset before lowering the transmission power. Tielert et al. show that their algorithm is fair with respect to Pareto optimality and effectively reduces inter-reception times, even for groups with different target distances. However, use cases building upon awareness provided by beacon exchange may require diverse target distances. Furthermore, calculation of the optimal transmission power presumes a (deterministic) mapping from target distance to transmission power. Actual radio propagation conditions are rather unpredictable, which puts a question mark over such a 'distance to power' mapping function. Under heavy interference, transmissions using too low power will not achieve the target distance. Overly conservative mappings, *i.e.* those with considerable margins to protect against interference, introduce significant interference on their own. This conjuncture defeats the initial purpose of TPC.

Kloiber et al. [117] avoid the mapping of range to transmission power to elude the imponderables coming along with it. Instead, they opt for random transmission powers, which sounds ineffective at first. The aim of their Random TPC is not to harmonise power levels among stations but to counteract correlated packet collisions and evading the dilemma to decide for a single power versus rate trade-off. Kloiber et al. argue that in some traffic situations, *e.g.* driving on a highway, the relative position of vehicles to each other is quasi-static, *i.e.* their velocities are almost equal, and thus distances do not change a lot. Likewise, these roughly equal vehicle dynamics trigger the generation of CAMs at the same rate within these vehicle groups. Packet collisions may thus occur in bursts due to a regular transmission pattern and steady radio propagation conditions. Not receiving multiple CAMs in a row is more severe, though, because it reduces the awareness about other vehicles due to increased inter-packet intervals. Frequently varying the output

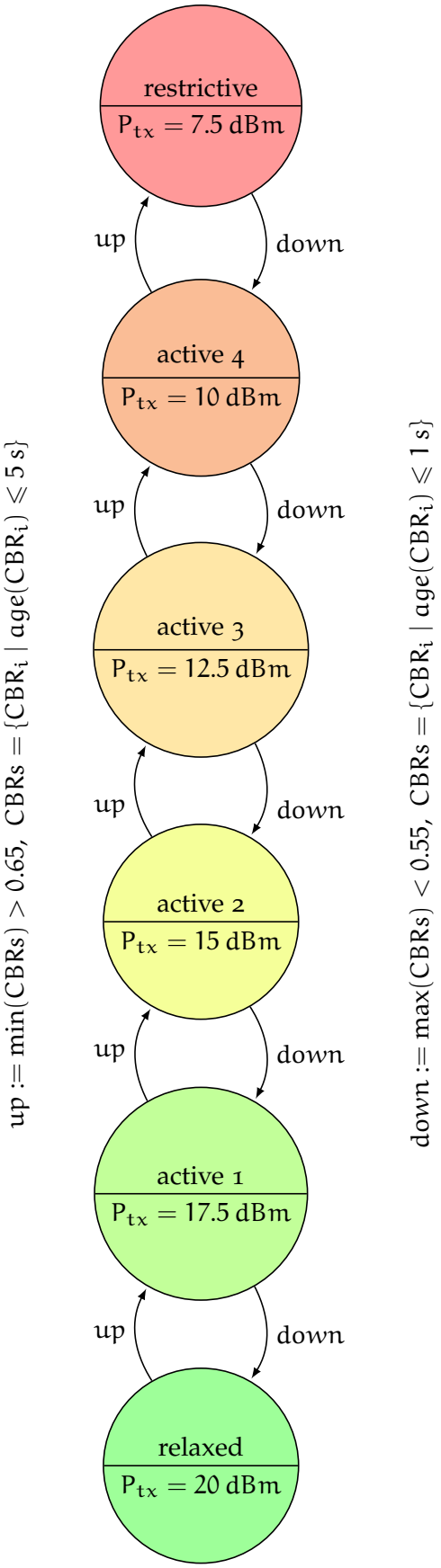


Figure 2.9: TPC by Subramanian et al. [79] using a reactive DCC state machine

power of CAM transmissions implicitly affects the inter-packet intervals as perceived by receivers. While distant vehicles will receive only high-power transmissions, nearby vehicles receive low- and high-power transmissions. This effect is called 'spatial awareness' by Kloiber et al.

Many TPC strategies, *e.g.* joint TPC+TRC by Tielert et al. [93] or Distributed Fair Power Adjustments for VANETs by Torrent-Moreno, Santi and Hartenstein [41], determine for a given rate and channel load a single power level deemed as optimal. Random TPC [76, 117] finds an optimal power distribution instead. Selection of distribution parameters, *i.e.* shape, mean and variance, can be tuned to achieve a particular 'spatial awareness'. Simulations by Kloiber, Härri and Strang [76] employed uniform distributions from 3 dBm to 33 dBm, *i.e.* all ranges are treated equally important. If all transmitters use power distributions with equal means, statistical fairness among the stations is achieved because none drowns out others on average. The variance counteracts the correlation of packet collisions spatially.

Independence of any feedback by receivers as wells as likely wrong path loss estimations are attractive features of Random TPC. It is also not optimised for a specific use case focussing on a particular range but trades off more short-ranged against less far-reaching transmissions. This compromise is acceptable as critical driving situations usually involve only nearby vehicles. Remote vehicles can still be tracked via their CAMs, though at a reduced update rate. On the downside, Random TPC addresses only periodic, awareness-related message types such as CAMs. Event-triggered messages and multi-hop packet forwarding are not even considered. Kloiber et al. have only evaluated uniform power distributions in [76, 117]. Possibly, other distributions fit VANET communication better.

Bai, Stancil and Krishnan [56] measured packet delivery ratios under various channel conditions and for power settings of 10 dBm, 15 dBm and 20 dBm. Measured packet delivery ratios for single-hop communication suggest 15 dBm as a good trade-off between communication success and spatial reuse of the channel for communication up to 300 m. While 10 dBm tends to be too weak for a clear signal above background noise at the receiver, 20 dBm is only a minor improvement. Bai et al. attribute this observation to interference becoming the dominant distortion. To the best of my knowledge, neither ETSI nor C2C-CC suggests a default transmission power for CAM transmissions. As long as the used transmission power conforms to the aforementioned limits from 3 dBm to 33 dBm, any value may be chosen at discretion.

If not using a constant power setting, randomising power for CAM transmissions is the most convincing approach at the moment. It is not optimised for a particular use case building upon CAMs and does not rely on any power-distance mapping, which cannot cover the vast number of channel

Table 2.6: MCSs for ITS-G5 as given in [121, Table 17-4]

Data rate (Mbit/s)	Mandatory	Modulation	Coded bits (per OFDM symbol)	Coding rate
3	•	BPSK	48	1/2
4.5		BPSK	48	3/4
6	•	QPSK	96	1/2
9		QPSK	96	3/4
12	•	16-QAM	192	1/2
18		16-QAM	192	3/4
24		64-QAM	288	2/3
27		64-QAM	288	3/4

conditions. A possible future improvement could be the combination with a TPC algorithm finding the optimal power level for a given channel load and vehicle density: Instead of applying this power level constantly for all transmissions, it may be employed as the current mean for variable power distributions. Considering the finding of Bai, Stancil and Krishnan [56], it may be beneficial to assign power levels in the range between 15 dBm to 33 dBm lower weights, *i.e.* use a random distribution with a flattening tail. Hence, the variability of power levels is pronounced in the low-power range where it comes into play most.

Switching Modulation and Coding Schemes

IEEE 802.11 [121] defines a set of MCSs available for 10 MHz channels in the 5.9 GHz band, as they are employed by ITS-G5 [86]. The modulation – Binary Phase Shift Keying (*BPSK*), Quadrature Phase Shift Keying (*QPSK*) or a Quadrature Amplitude Modulation (*QAM*) variant – determines the number of bits that can be coded per time unit, *i.e.* per length of one OFDM symbol. The net data rate of a transmission is the combination of a modulation with a particular coding rate, the ratio of data bits to coded bits. The remaining bits are employed for error correction. Table 2.6 summarises the MCSs along with their main properties and resulting data rates. However, only three MCSs are mandatory to be decodable by compliant radios.

The higher the selected data rate for a packet transmission, the better the SNIR needs to be for a successful reception. Consequently, the achievable range is the main limitation of high data rate MCSs. According to Bazzi et al. [123], it is impossible to bridge a distance of 400 m using a rate of 27 Mbit/s.

Bai, Stancil and Krishnan [56] found the MCS of 6 Mbit/s to outperform the usually more robust 3 Mbit/s scheme as well as 18 Mbit/s, which is more sensitive to noise as expected. They attribute the poor performance of slowest data rate to the lack of dynamic equaliser in the employed radios, *i.e.* the

packet duration exceeded the channel coherence time. Radios designed explicitly for automotive usage may perform better than the employed consumer hardware, though.

Jiang, Chen and Delgrossi [49] simulate reception probabilities for various data rates and packet sizes, incorporating empirically determined SNIR thresholds for each data rate. Their findings also emerge 6 Mbit/s as the best performing data rate in a wide range of scenarios.

Karoui, Chalhoub and Freitas [199] employ the Artery simulation framework to evaluate their TDC, which switches data rates in addition to TRC by LIMERIC. Instead of using a fixed target CBR, they adapt the target CBR employed by LIMERIC according to the expected reception success of packets sent with a particular data rate. However, as the author admit, their required estimation of the link quality neglects varying interference and hidden stations entirely.

As outlined by Tielert [100], throughput is not a major concern in a VANET meant to increase traffic safety. Hence, it has become an established, safe choice to stick with 6 Mbit/s for single-hop broadcast communication. This is also the default rate specified to the G5-CCH channel in [86].

When channel load is high, though, switching to a higher data rate may be beneficial for multi-hop communication. For one thing, triggering conditions for DENM use cases as defined by C2C-CC incorporate timed packet repetitions [169]. For another thing, the CBF packet forwarding scheme of GN repeats packets while disseminating them in their respective destination area [193]. Possibly, a trade-off between packet redundancy for reliability and reduced channel occupancy can be found. For example, a subset of those redundant transmissions may be sent at a higher data rate.

Gathering Channel Busy Ratio Measurements

No matter if reactive or adaptive DCC mechanisms are employed, CBR measurements are the most important input parameter reflecting the prevailing the channel load. Though the way how to calculate CBRs has been defined by standardisation as outlined in Section 2.1.4, the timing of CBR measurements can make a difference.

By default, IEEE 802.11 radios in a VANET do not synchronise their clocks. Consequently, it is reasonable to assume that the CBR measurements reported by those radios are not aligned in time between distinct stations either. However, CBR measurements may get synchronised unintentionally when a DCC algorithm is demanding synchronised updates, such as the adaptive approach of *TS 102 687*. Especially if a particular radio implementation reports CBRs not via notification callbacks but only on explicit request, polling CBR measurements within the DCC update loop may lead to synchronised measurements.

Bansal et al. [95] observe severe oscillations of reported CBRs in their simulation of reactive DCC adapting transmit rate by five states. These oscillations are attributed to clustering of CAM transmission time points. Since transmissions are then not evenly spread over time, the risk for packet collisions is considerably higher as well. Additionally, Bansal et al. show that CBR oscillations are particularly pronounced if measurement intervals are synchronised among stations. Oscillations can be reduced by 40 % to 70 % with asynchronous measurements, but are still distinct with reactive DCC. The same group of authors dwell deeper into the observed clustering effect in [119]. In contrast to oscillations of the reactive DCC state machine itself observed by others [85, 81], the CBR oscillations have another reason. In fact, the state machine is continuously in the restrictive state in this case. Since only a discrete number of message rates is available with reactive TRC, vehicles tend to generate their CAMs in quite regular patterns, also in relation to each other. Synchronised CBR measurements amplify this regularity because rate changes are then synchronised as well. Rostami et al. [119] propose to substitute the discrete *active* sub-states by a continuous, linear rate function to break up the regularity. Continuous rate changes along with asynchronous CBR measurements avoid the undesirable clustering of message transmissions.

Simulations, where a global clock is readily available across all simulated entities, are particularly prone to artificial synchronisation. With the aforementioned observations in mind, the radio model employed for simulations in this thesis can generate CBR measurements synchronously and asynchronously. More details with respect to CBR reporting are given later in Section 3.2.3.

2.4.3 With a little Help from my Friends: Networked DCC

Standardisation of GN is divided into two parts: A larger, radio-agnostic document *EN 302 636-4-1* [193] and an accompanying document with specific extensions when GN is operating on ITS-G5 radios*TS 102 636-4-2* [195]. A considerable share of *TS 102 636-4-2* deals with information sharing in support of DCC. Up to now, the CBR measured by the local station has been the primary input for DCC algorithms. Since no rationale behind information sharing is given in *TS 102 636-4-2*, this section presents the original research work in this area, which influenced the later standardisation process.

Sharing CBR Measurements

The work on sharing CBR measurements among vehicles by Tielert et al. [70, 100] had an undeniable impact on standardisation by ETSI. The motivation behind sharing local CBR measurements is to reduce the impact

of local measurement errors. Information sharing helps to establish a common perception of the network's load. DCC algorithms distributed over vehicles then operate on a common data basis, which avoids imbalanced or even unfair decisions by individual stations. PULSAR, the TRC algorithm proposed by Tielert et al., aims to reach a target CBR by adapting the beaconing rate similar to LIMERIC. Two aspects distinguish PULSAR from LIMERIC: First, the rate is adapted following an Additive Increase Multiplicative Decrease (*AIMD*) scheme. The authors also consider to replace it with LIMERIC's scheme, though. The more vital, second difference is the piggybacking of CBR measurements over two-hop distances. Tielert et al. argue that the carrier sensing range is about twice the transmission range. By propagating CBR measurements over two-hops, stations can detect the overall channel load. They say, a station then participates in congestion control not only for those stations it communicates with but also for those it interferes. PULSAR's piggybacking scheme expects each station to attach its *local* CBR to outgoing messages as $\mathrm{CBR}_{\mathrm{L}}^{0}$, as well as the maximum CBR reported by its one-hop neighbours as $\mathrm{CBR}_{\mathrm{L}}^{1}$. Each station can determine $\mathrm{CBR}_{\mathrm{L}}^{1}$ by choosing the maximum from the $\mathbf{CBR}_{\mathrm{R}}^{1}$ set, consisting of *received* $\mathrm{CBR}_{\mathrm{L}}^{0}$ values. In the same manner, $\mathrm{CBR}_{\mathrm{L}}^{2}$ can be determined from the $\mathbf{CBR}_{\mathrm{R}}^{2}$, the set of all received $\mathrm{CBR}_{\mathrm{L}}^{1}$ fields. A receiving station thus knows its local CBR and the maximum CBR for its one- and two-hop neighbours, $\mathrm{CBR}_{\mathrm{L}}^{1} = \max(\mathbf{CBR}_{\mathrm{R}}^{0})$ and $\mathrm{CBR}_{\mathrm{L}}^{2} = \max(\mathbf{CBR}_{\mathrm{R}}^{1})$ respectively.

Pros and cons of aggregating CBRs by their maximum versus mean are discussed in the PhD thesis of Tielert [100]. The maximum norm is a rather conservative option, and TRC will reduce the message rate already for a single congested station. This behaviour is also mentioned as a possible vulnerability of the algorithm because faulty measurements may cause unnecessary rate reductions. Though the mean is less affected by some faulty measurements, network-wide rates vary less with the maximum norm in the evaluated scenario.

As shown by Fig. 2.5 in Section 2.1.4, the standardised SHB header includes the 'DCC-MCO' field in ITS-G5 networks, which conveys $\mathrm{CBR}_{\mathrm{L}}^{0}$ and $\mathrm{CBR}_{\mathrm{L}}^{1}$ of a station to its neighbours. Adaptive DCC, as described in Section 2.1.4 and Section 2.4.2, smoothes the raw CBR measurements CBR_{raw} reported by the radio hardware as per Eq. (1). As a recapitulation: The two latest raw measurements are averaged and constitute the recent observation. The updated smoothed CBR_{local} is then the moving average of this recent observation and the previous CBR_{local}. With implemented information sharing, the locally measured CBR_{raw} values of Eq. (1) are substituted by CBR_{global} values, which represent the global view on channel congestion. Similar to CBR_{raw}'s update frequency, CBR_{global} is calculated every 100 ms, according to Eq. (10).

$$\mathrm{CBR}_{global}(n) = \max\{\ \mathrm{CBR}^0_L(n-1),\ \mathrm{CBR}^1_L(n),\ \mathrm{CBR}^2_L(n)\ \} \tag{10}$$

Noteworthy, *TS 102 636-4-2* [195] diverges from the original PULSAR concept by Tielert on some points. ETSI refers to all received 'DCC-MCO' fields within the last second as the elements of $\mathbf{CBR}^0_R$ and $\mathbf{CBR}^1_R$. PULSAR instead restricts $\mathbf{CBR}^0_R$ to those CBR^0_L received between update intervals $n-1$ and $n-2$. Likewise, $\mathbf{CBR}^1_R$ consists of the CBR^1_L fields generated two cycles before but received recently. This filtering requires timestamps describing the age of included CBRs, which exist in PULSAR's piggybacking header but not in ETSI's 'DCC-MCO'. A leftover of PULSAR's sophisticated effort to compensate dissemination delays of its CBR piggybacking can be found in Eq. (10): The equation considers the local CBR^0_L from the previous time interval, but one- and two-hop CBRs from the current interval. As a further simplification, the piggybacking header of *TS 102 636-4-2* also does not convey the station's transmission rate as PULSAR does.

Despite the 'global' naming of CBR_{global}, every station calculates its own view on the (assumed) global CBR, which is expected to be practically the same for all nearby and interfering stations. Autolitano et al. [94] look for correlations between local, one-hop and two-hop CBRs and number of neighbouring vehicles based on prior work by Tielert et al. [70]. They take the obstacle shadowing by buildings into account, so only vehicles at intersections on the studied grid road network are exposed to two LOS paths. Surges of directly reachable neighbours in intersections correlate with peaks in the locally measured CBR. Only minor differences between one-hop and two-hop CBRs are found, though. Both are not affected by the number of directly or indirectly (one-hop) neighbours but rather stable. In fact, the two-hop CBR is equal among all stations on the employed $750 \times 750\,\mathrm{m}^2$ grid map. Autolitano et al. point out the necessity of further closed-loop simulations which consider the mutual influence of DCC and CBR. They also assume obstructions by vehicles and less regular map topologies may pinpoint additional characteristics of CBR calculation methods. Since only single-hop packet transmissions have been considered by Autolitano et al., a possible influence of multi-hop packet communication on n-hop CBR metrics remains unstudied. Chapter 5 of this thesis gives more insights into the accuracy of CBRs in sophisticated environments.

Conveying Transmission Power of Packets

Aygun, Boban and Wyglinski [112] propose another DCC strategy coined as ECPR, which takes the environment as well as the traffic context to apply TRC and TPC into account. TRC of EPRC is essentially LIMERIC and TPC is designed to reduce channel load jointly for better spatial reuse of the channel. From this perspective, ECPR does not differ significantly from other

TRC+TPC algorithms [93, 117]. These algorithms boil down to finding a compromise between higher transmission power and reduced packet rate, or vice versa. ECPR, however, actively determines the signal propagation characteristics of its current environment. It does not rely on any a priori knowledge such as maps but depends upon the cooperation of other stations.

ECPR's TPC aims at selecting a transmission power just high enough to reach a given percentage of vehicles (target awareness) within a given range (target awareness range). For this purpose, all messages received from others during the last measurement period (200 ms are assumed in [112]) are evaluated, if they have not been further away than the target distance. Beside typical fields such as position, these messages are expected to convey their original transmission power as well. With these prerequisites, the receiver can calculate the distance and path loss for each transmission. While path loss is the mere difference of transmission power and reception power, it can be transformed to a path loss exponent, which does not depend on the distance but describes the amount of attenuation for the given link. Aygun, Boban and Wyglinski [112] use the generic log-distance path loss model [29] for this transformation. After having a set of path loss exponents, the path loss model is turned around to calculate the necessary transmission powers. From these transmission powers, the percentile matching the given target awareness is ultimately selected. Consequently, this TPC mechanism will not necessarily pick the highest transmission power but a power level high enough to reach a sufficient portion of stations. Under harsh conditions, *i.e.* when the share of high path loss exponents increases, this TPC assigns higher power levels for the same distance.

Showing the advantages of ECPR by simulation requires a radio propagation model, where signal attenuation varies significantly because of shadowing effects. These variations need to be deterministic because ECPR relies on equal attenuation in both directions of a particular link, also called channel reciprocity [29]. Aygun, Boban and Wyglinski [112] evaluate their ECPR algorithm using GEMV² for this reason, which will be covered in Section 3.2.1.

Instead of assuming a fixed distance/transmission power relationship [79, 116, 93], ECPR incorporates the actual, measurable path loss conditions. Such measurements presuppose other stations to include their transmission power in their outgoing messages cooperatively. ECPR has been incorporated by ETSI in *TR 101 613* [104] and the necessary piggybacking of transmission power is part of the SHB header outlined in standardisation [195]. However, the LocTEs only keep book of the last packet reception's Received Signal Strength Indicator (*RSSI*) and accompanying transmission power. Though GN is prepared to piggyback transmission powers, it does not prescribe any

particular TPC algorithm. Aygun, Boban and Wyglinski [112] explicitly note that ECPR shall only govern the power of awareness messages such as CAMs but not of safety-critical messages such as DENM generated due to a hazard.

Inclusion of the transmission power in the outgoing SHB header is problematic in general. While fields such as CBR measurements are already 'facts', the transmission power is only going to become a fact. If the access layer does not or cannot adhere to the transmission power given by the network layer, power calculations at the receiver side are inherently wrong. For example, the access layer may be unable to transmit at a given power level when the station is currently subject to power regulations within a 'protection zone'. Setting the encoded transmission power of the GN layer's field at the access layer is no option either. Despite the violation of layer isolation, this would also break the cryptographic signature. An enhanced DCC cross-layer could provide a possible remedy of this issue: If GN asks for a suitable power level at DCC, then DCC has to reply either the maximum allowed power or the optimal power according to its TPC routine, whatever is lower at the moment. DCC needs then to guarantee that the access layers also applies the previously indicated power level.

Interaction with Packet Forwarding

Beyond the piggybacking of information, the network layer itself needs to interact with DCC. Bellache, Shagdar and Tohme [124] claim that applying the same DCC mechanisms suitable for periodic CAMs to multi-hop packet forwarding may be self-defeating. Reducing transmission power of multi-hop packet may result in an increased overall channel load because it needs to be forwarded more often to cover the same area. As an alternative, Bellache et al. propose to adapt the retransmission counter found in CBF depending on prevailing channel load. They suggest to keep the retransmission counter $RC_{th} \leqslant 4$ because prior studies found further retransmissions contribute only marginally to the coverage. Whenever updated CBR measurements are available, the retransmission counter gets adjusted: If the current CBR is below 55 %, RC_{th} is stepwise increased up to 4 at maximum. Vice versa, if CBR is above 70 %, RC_{th} is decreased by one until the lower bound of 1 is reached. Otherwise, the retransmission counter remains unchanged. Simulations by Bellache, Shagdar and Tohme show improved packet delivery ratios for CAM and DENM traffic when the amount of retransmissions is reduced in dense vehicle scenarios. In such dense scenarios, the reduced network load outweighs the risk of interrupted forwarding by single forwarders. Unfortunately, the approach does not work well with any TRC aiming for a stable target CBR, *e.g.* LIMERIC.

Bellache, Shagdar and Tohme [125] further enhance the CBF algorithm by not only adapting the retransmission counter but also the waiting time for a packet in the forwarding buffer. The motivation for this approach is twofold: Reducing the forwarding delay by faster-scheduled retransmissions and a reduced packet collision risk. Latter is due to nearby stations determining practically identical waiting times because their distance to the sender is almost the same. Hence, their retransmissions are also scheduled at the same time with an exacerbated collision risk. However, shrinking the waiting time for low channel loads and scaling it up at high CBRs is again only viable with reactive but not adaptive DCC.

The previously cited works [124, 125] incorporate information provided by DCC into GN, but Kühlmorgen et al. [200] also see the necessity to make DCC explicitly aware of GN's forwarding behaviour. They specifically investigate the interaction between a TRC gatekeeper based on LIMERIC and multi-hop communication using the standardised CBF. In this setup, the message rate advised by LIMERIC is applied to all outgoing traffic, *i.e.* DENM and CAM packets with three distinct priorities in total. As a consequence, packets may get queued by DCC if the current rate limit is reached and even dropped if the queue capacity or maximum queuing time is exceeded. In congested scenarios, when the queuing durations rise, this breaks the overhearing mechanism of CBF, which is meant to prevent broadcast storms: As soon as DCC enqueues a CBF packet, GN cannot stop this transmission anymore even if it has 'heard' retransmissions by others. Way too many retransmissions are thus injected into the VANET, wasting already scarce channel capacity and ultimately deteriorating the coverage of CBF.

Kühlmorgen et al. [200] propose to add another CBF duplicate check to the gatekeeper. This sub-system in the gatekeeper named RORA shall restore the overhearing of CBF retransmissions, *i.e.* discard own forwarding actions when the station receives the same packet by others meanwhile.[2] For this purpose, it queries the duplicate packet list maintained by CBF in the network layer. If the current packet is detected to have become a duplicate in the meantime, *i.e.* between CBF handling by GN and end of DCC queuing, it will be dropped by the gatekeeper. Though Kühlmorgen et al. show their approach only for simple CBF, *i.e.* without retransmission counters > 1, it should be trivial to extend RORA in this regard. For example, instead of checking for the existence of a duplicate entry, this check may get generalised as 'is the forwarding of this packet still advised?' query towards GN. This enhancement would relieve the gatekeeper from needing to know details about the configured forwarding algorithms in GN.

2 The RORA mechanism has been added as an informative appendix of *TS 102 636-4-2* [195].

As a side note, Kühlmorgen et al. [200] also compare two DCC queueing policies, Simple Priority Queuing and Weighted Fair Queuing. Former causes entire starvation of low-priority traffic while multi-hop coverage is sensitive to the particular weights chosen for latter policy. Their observation underpins the previously stated necessity for proper queue management to achieve QoS goals in VANETs, see Section 2.3.

2.4.4 Congestion Control at Source: DCC Facilities

A gatekeeper at the link layer has definitely its right to exist, even if it is just deemed as the last line of defence against excessive transmissions. However, such a gatekeeper can only drop packets or delay their transmission time points. While the former causes wasted efforts by upper layers, the latter might be unacceptable for services with low delay tolerance. When no channel resources are available at the moment, it is thus reasonable to omit the generation of a message at the source. In ITS-G5 stations, the services located at the Facilities layer are the source for new messages. This section examines the Facilities layer's relations with congestion control.

Timing of Facilities' Actions

The CA service is a prominent example of a 'message generator' that wants its messages on-the-air as quickly as possible. Instead of transmitting a CAM with slightly outdated information due to queuing, this service prefers to postpone message generation in favour of low latency. Hence, the CA service can query the currently allowed, shortest packet interval T_GenCamDcc from DCC [171]. While this approach avoids the generation of CAMs when the gatekeeper would drop or enqueue them due to TRC constraints, an undelayed transmission is not guaranteed either. In fact, a race condition may occur in the time from querying T_GenCamDcc until the CAM reaches the gatekeeper. Packets transmitted in the meantime by other services may elongate the demanded gap between consecutive CAMs. A more elaborated interlocking between DCC transmission constraints and services could solve such issues.

Besides throttling of its message rate because of DCC constraints, CA services will also adapt their message rate according to their host vehicles' dynamics. This voluntary self-restriction avoids transmission of CAMs with only minor novelty compared to the previous CAM, *i.e.* neither position, speed, nor heading of the vehicle have significantly changed since. Even without explicit rate control by DCC, a CA service's message rate is in the range 1 Hz to 10 Hz. The need to 'stay visible' even when the vehicle is standing still motivates the lower bound. Higher rates become necessary when receivers' assumptions about a transmitter's location at some point

of time would be too vague otherwise. CA specification [193] names fixed divergence thresholds of 4° for heading, 0.5 m/s for speed, and 4 m for location displacement. These settings have been agreed upon to strike a balance between mutual awareness and channel load.

Optimisation of the beaconing rate, *i.e.* the CAM rate in an ITS-G5 network, has drawn much attention by academia over the years. Without claims of being complete by any means, a few ideas are worth mentioning paradigmatically. Bansal et al. [84] propose with EMBARC to generate just as many beacon messages as allowed by the underlying LIMERIC algorithm, *i.e.* rate is not capped at 10 Hz if channel resources are still available. EMBARC is obviously unable to support any other service or packet forwarding because beacons deliberately consume all channel resources. Notably, EMBARC also explicitly borrows *information sharing* from PULSAR to enhance LIMERIC.

Sepulcre et al. [99] also build upon LIMERIC and PULSAR. However, they assume that stations have individual requirements for a minimum message rate. Stations will thus generate beacons at their demanded rate plus an adaptable margin. In contrast to LIMERIC, it is not the rate but this margin that will be adjusted according to the channel load. Thus, stations in the vicinity will not converge to a common message rate but a common margin which is generated on top. Piggybacking by PULSAR is also modified to share not only CBRs but also the employed beaconing margins. Again, this mechanism cannibalises resources to the detriment of other services.

Hajiaghajani and Qiao [178] present an alternative approach based on game theory to adapt the message rate of beaconing. Vehicles are considered as 'selfish players' aiming at maximising their utility function, which is continuously increasing with the chosen message rate. The utility function consists of a gain term, representing the chance to avoid a rear-end collision, and a cost term penalising transmissions with growing channel congestion. Thus, safety and fair channel access are taken into account by this utility function. When facing a congested network, a vehicle will only increase its message rate in risky situations, *i.e.* the costs are justified by more safety. Once more, only a single use case (rear-end collision avoidance) is covered by Hajiaghajani and Qiao's rate adaptation. Other use cases and services are not addressed by them at all, not to speak of packet forwarding. Their approach loses quite some attraction if all these aspects need to be incorporated in a utility function. It may be even impossible to find a suitable function for a network where stations have different interests and capabilities.

Thandavarayan, Sepulcre and Gozalvez [202] adopt CA's concept of checking TRC limits before message generation for CP. They prefer to include detected objects in a CPM only if they have changed since the last message, similar to the CAM triggering conditions. Hence, their CPMs tend to get longer in size as the interval of consecutive CPMs increases. When CP

considers the DCC rate constraints, the update rate is considerably lower. However, enqueuing its packets is also avoided, and thus the average information age is cut down. Simulations by Thandavarayan, Sepulcre and Gozalvez show an improved awareness ratio of perceived objects, thus.

Zhu, Goswami and Li [186] examine the adaptability of a platooning application towards DCC's TRC limits. They still note an advantage of the platooning application over plain ACC even when TRC reduces the update rate of platooning messages down to 1 Hz. If a certain rate cannot be maintained any longer, *e.g.* due to increasing channel congestion, a platooning application can adapt to this by expanding the headway to the preceding vehicle. Since physical movement is involved in this process, such an application would benefit from announcements by TRC about upcoming rate changes in time.

Controlling Message Length

Services have multiple options to reduce their share in channel resource usage. Except for omitting generation of a message altogether, a service may also control the message length by skipping optional data or limiting the number of encoded elements in lists. For example, CAM's so-called 'low-frequency container' is only included once every 500 ms [171]. Similarly, CPMs may only incorporate a subset of objects to mitigate redundancy of reported objects among stations [175].

Data compression is a complementary approach to reduce message length without sacrificing informative content. Sepulcre, Tercero and Gozalvez [153] have studied the gain of This approach can reduce the message length by 4 % to 14 %, where initially longer message tend to lead to higher compression gains. However, compression by transmitter and decompression by receiver implies further processing costs, which manifest as increased latency in the order of a few milliseconds.

Rufino et al. [182] underline the necessity to consider the varying sizes of CAMs; especially, the variable overhead introduced by securing these messages with signatures and certificates must be considered. In extreme cases, when a CAM itself is short but full security certificates are attached, up to 80 % of the entire packet size can be attributed to secured message headers. Even for large CAMs, over 60 % of the total size is still caused by security headers. The network layer attaches these secured message headers, *i.e.* Facilities cannot control if a particular packet will convey a full certificate or only its digest. As a consequence, if DCC tracks services' channel usage, it should observe the payload lengths passed to the network layer because the additional overhead is outside services' reach.

According to standardisation *TS 103 097 - Intelligent Transport Systems (ITS); Security; Security header and certificate formats*, certificate digests are only allowed for CAM packets while DENMs are always accompanied by a full certificate [106]. Feiri, Petit and Kargl [74] have compared various schemes omitting certificates to reduce the packet size. These schemes can be based on elapsed time, changes in the set of neighbours, or channel congestion. Standardisation favours to add a full certificate to a CAM when a message from a previously unknown station has been received, *i.e.* the set of neighbours has changed. When the certificate of a station is not known yet, its CAMs need to be discarded by those receivers, resulting in a so-called cryptographic packet loss. Certificate omission schemes thus have to weigh up the risk of discarded packets to larger packet sizes. Feiri, Petit and Kargl propose a congestion-based omission scheme that adapts the occurrence of full certificates according to the number of neighbours. Strictly speaking, such a scheme belongs rather to a not-existing DCC_SEC entity.

Orchestration of Services

Despite the explicit adoption of TRC constraints in the CA service, standardisation of DCC at Facilities is still in progress, though, *i.e.* only an unpublished draft of *TS 103 141* exists [198]. Allocating channel resources to particular services is the key feature of DCC_FAC, especially when many services compete for resources, insufficient to fulfil all desires. The draft mentioned above suggests to monitor the average channel usage per application and traffic class and assigns currently available resources based on this prior usage share. Though this approach does not anticipate uniform message lengths and occurrences per service, it favours those services occupying the channel more often. Collaborative services refraining from channel usage in favour of other services are thus penalised unwarrantedly.

Furthermore, Khan and Härri [148] point out that DCC_FAC from the standardisation draft is incompatible with reactive DCC_ACC, *i.e.* TRC based on a state machine. DCC at Facilities layer assumes that it can distribute the station's channel resources at will among services. However, reactive DCC's gatekeeper considers only the arrival of a packet but not its effective airtime. This behaviour defeats any strategy at the Facilities layer that may increase the packet rate in favour of shorter packet lengths as each transmission must follow the same T_{off} inter-packet pause.

Khan and Härri [147] break up the strict T_{off} transmission pauses between packets in favour of more flexible quotas. For each TC a quota is managed by each station, which denotes the remaining transmission opportunities. A packet can then be transmitted without delay as long as the quota is sufficient for the packet's TC. In case the current quota is insufficient, the packet is enqueued and transmission thus deferred. Quotas are filled up again

periodically, *e.g.* as part of the DCC update cycle re-calculating the station's allowed channel usage. Beside carrying unused quota over to the next cycle, a service may also borrow from a future quota to some extent. Additionally, quotas can be re-balanced within a station among TCs: If the quota of a higher TC is insufficient, assigned quotas of lower TCs get reduced in the next cycle. Starvation of low TCs is avoided by 'stealing' quota from the next higher TC after a maximum amount of deferral. Services can further on adapt their packet generation rates by evaluating available quotas. Simulations by Khan and Härri employing three services using distinct TCs promise generally reduced transmission delays in networks with 100 to 240 stations. Notably, one of these services generates packets as bursts, which are impossible to transmit with low delay under a strict T_{off} transmission scheme. Care must be taken, however, if several services are waiting for transmission grants at the end of rather long quota cycles. Khan and Härri suggest a cycle duration of one second. Allowing all pending services to generate their packets at the beginning of a new cycle is prohibitive as it would cause artificial bursts. While flexible resource usage is clearly a strength of the presented quota mechanism, introducing fixed cycles looks like an unfortunate design decision.

Considering the increasing variability of V2X services, relying on plain TC for prioritisation among those heterogenous services may be too simple. The actual communication need of a service may vary over time as it adapts to channel congestion and changing traffic situations. Khan, Sesia and Härri [179] propose a dynamic orchestration of channel resources among services based on a station's channel usage limit, which corresponds to δ of the LIMERIC algorithm. This orchestration needs to take place at Facilities layer because lower layers are unable to differentiate services having equal TCs. Two aspects of their contribution are noteworthy: On top of the static priority levels, they add *urgency* and *usefulness* as dynamic factors when determining the channel usage share δ_i of each service i. While *urgency* favours services with a short deadline, *usefulness* takes a message's anticipated value for its recipients into account. Furthermore, Khan et al. have overcome some issues of their prior quota approach [147]. Instead of granting every service its transmission requests as long as it has a sufficient quota, the control is inverted here: DCC_FAC notifies the service with the highest transmission budget to generate its message. Hence, this approach orchestrates resource usage among services and prevents the generation of packets while DCC_-ACC forbids any transmissions.

When a service requests a transmission, its budget needs to be sufficient to cover the associated costs. The costs C_i for a transmission grow with increasing airtime T_{on} in relation to pause T_{off} between a service's transmissions as per Eq. (11). On the income side, a service's budget increases

continuously by accumulated resources A_i with increasing duration Δt since its last transmission as per Eq. (12). The larger a service's channel usage share δ_i, the faster its budget recovers and the less it needs to expense.

$$C_i = \frac{T_{on,i} * (1 - \delta_i)}{T_{off,i} * \delta_i} \tag{11}$$

$$A_i = \frac{\Delta t * \delta_i}{T_{on,i}} \tag{12}$$

Since all budgets with $\delta_i > 0$ grow over time, even low priority services will not starve. Unfortunately, Khan et al. do not outline how to initialise their algorithm. At the very beginning, neither the duration of a prior transmission $T_{on,i}$ nor the gap between transmissions $T_{off,i}$ exist, which are essential for budget updates. Budget growth is also potentially infinite, which is not a problem in the simulation by Khan et al. because all services have recurring transmission demands. However, services transmitting only now and then may accumulate excessive budgets and thus essentially block other services. Related to this issue, it remains unclear how DCC_FAC shall proceed if the service with the highest budget does not want to transmit anything at the moment.

2.4.5 Congestion Control in other Networks

Speaking about congestion control, one may first think of the mechanisms linked with TCP [52]. Since TCP is a connection-oriented transport protocol, it aims to utilise available bandwidth between sender and receiver as much as possible. Receivers will acknowledge reception of data explicitly, enabling the sender to detect the amount and rate of data the network and ultimately the receiver can handle. *TCP Congestion Control* defines only a framework for congestion control in TCP, *i.e.* the action scope of TCP algorithms implementing congestion control. Thus, many flavours have been developed over the years, such as 'NewReno' [75] and 'CUBIC' [48], the current default in Linux kernels[3]. Although TCP's congestion control is an exciting and active research domain, its challenges are very different from congestion control in VANETs. Foremost, one-to-one data connections are a sporadic IVC use case while it is the Internet's workhorse. Jiang et al. [40] further point out that packet loss in TCP networks is predominantly caused by congestion, but this is not true for the less reliable, wireless communication links in a VANET.

Sadeghi deals with congestion control in wireless mesh networks [55]. In an 802.11 mesh network[4], packets are forwarded via multi-hop whereas in traditional 802.11 networks – with one access point and multiple clients

3 Checked in Linux 5.5 source code
4 Wireless mesh networks as introduced by the amendment IEEE 802.11s

– single-hop communication dominates. Network throughput in a mesh network is prone to packet collisions because a single collision along the forwarding path equals a total loss and channel usage by previous hops has been wasted. Hence, avoiding congestion and overload of forwarding nodes are major concerns in these mesh networks.

Contrary to measuring CBRs in a VANET, detection of congestion in a wireless mesh network relies on observing the queue length at the MAC layer. When a mesh node detects congestion as its queue builds up, it may ask neighbouring nodes to reduce their channel access and upstream nodes not to forward their packets to this node. IEEE 802.11s thus designates explicit signalling of congestion via its mesh beacons to mitigate performance issues. These signals indicate the estimated duration of congestion so nodes can plan accordingly. However, no specific actions or congestion resolution protocols have been standardised. [121]

Though topology of mesh networks is rather static compared to a VANET and end-to-end flows are prevailing instead of broadcasts, the physical challenges are very much the same. While the image of pipes with a fixed capacity works for wired network links, the variations found at the physical layer of wireless networks demand special treatment. In this context, Yi and Shakkottai [47] present an approach to spread the load over space: Every node runs a local congestion controller and reports its congestion level to the upstream node for each individual flow. With increasing congestion in one region of the network, it becomes more attractive to avoid busy downstream nodes in such a region and forward packets taking a detour. Only local information is necessary to decide for the next hop, but the authors have shown that the spatial spreading of network load can be realised this way.

For the US alternative of ITS-G5 building upon Basic Safety Messages (*BSMs*), SAE International's standard *J2945/1* outlines how congestion control shall be realised for their environment [120]. The BSM format is designed to cover both, periodic and event-based use cases, by adding the event-specific data as additional data container. Consequently, congestion control is very much equal to BSM scheduling, especially as no message forwarding is foreseen either. BSMs are generated every 100 ms to 600 ms; however, the channel load has no impact on the message rate but the density of vehicles nearby is considered. Critical messages will be scheduled earlier and transmitted with a higher priority, *i.e.* Access Category (*AC*) *Voice* instead of the normal *Video*. Channel load is an input variable for the transmission power, though. Up to a channel load of 50 %, BSMs are transmitted at 20 dBm. Beyond this load level, the transmission power will be gradually reduced down to 10 dBm, except for critical messages and those triggered by vehicle's dynamics. Channel load is measured quite similar to ITS-G5: Raw measurements are the percentage of the busy time in terms of carrier sensing

and energy detection during the last 100 ms. Those raw measurements are further smoothed by a moving average calculation before passed on as input variable for TPC. No TDC is involved, *i.e.* all BSMs are transmitted at a data rate of 6 Mbit/s. Though the US counterpart of a VANET employs the same radio technology as ITS-G5, its congestion control cannot be applied to ITS-G5 due to the aforementioned differences at upper layers.

Yoon and Kim [204] apply the congestion control of *J2945/1* to Cellular Vehicle-to-X (*C-V2X*). They find the TPC to be rather ineffective because TRC already throttles channel usage severely before power reductions apply. In fact, even a scenario with extremely high vehicle density (222 vehicles per lane and kilometre on a six-lane highway) triggers a power reduction by less than 10 %. They argue that a less aggressive TRC is favourable as it can reduce the update delay of safety messages for short-range links, i.e. vehicles separated up to 75 m. Yoon and Kim conclude that an exacerbated imbalance between TRC and TPC in *J2945/1'* congestion control exists when C-V2X is employed.

Bazzi [160] compares the effectiveness of TRC, TDC, and TPC in 802.11p and C-V2X sidelink communications using LTE. He concludes that TPC is ineffective and optimal TDC depends only on vehicle density in LTE-V2X, whereas all three mechanisms affect the trade-off between range, delay and congestion in 802.11p. With both technologies, TRC affects the inter-packet gap and thus the update delay. However, C-V2X is prone to exaggerate those delays because of its resource allocation.

Minstrel maintains a table of neighbours and MCSs employed on those links [205]. Depending on the success rate of previous transmissions, either a faster or a more robust, slower data rate is selected for upcoming transmissions. A single failed transmission does not immediately downgrade the data rate as link statistics are gathered by an Exponential Weighted Moving Average filter. In order to gather statistics, *i.e.* success probabilities, for all possible data rates, one-tenth of all data packets is transmitted with a randomly selected data rate. Obviously, this mechanism requires a reliable determination if a transmission has been successful. Even more, Minstrel prefers to retransmit a packet at a high rate instead of a single transmission at a lower rate as long as throughput is optimised. This approach is feasible without much effort if unicast frames are exchanged at the link layer, *i.e.* receptions are acknowledged by the receiver explicitly. Once more, the majority of packets in an ITS-G5 VANET are broadcasted and thus remain unacknowledged.

Hühn [90] presents a joint TDC+TPC algorithm named Minstrel-Blues. This algorithm aims to maximise throughput in a WLAN, *i.e.* with focus on unicast traffic. Similar to plain Minstrel, the Blues algorithm, which realises TPC, gathers transmission success statistics from unicast acknowledgements

by varying transmission powers. Over time, optimal power settings for each data rate are derived by this sampling method. However, sampling over longer periods is self-defeating in rapidly changing VANETs. According to Hühn [90], Minstrel-Blues requires 20 s to 35 s in a two-link setup to reach the optimal state and at least 50 s in a three-link setup.

One may ask if it is worth all the effort to 'fix' all the shortcomings of CSMA/CA instead of switching to another technology without these deficits. C-V2X divides the channel into resource blocks, which are spread over time slots and sub-frequencies. However, over distances of a few hundred meters, 802.11p performs reasonably well and is only surpassed by its cellular rival LTE-V2X at longer distances [123]. Though concurrent transmissions using distinct sub-frequencies will not collide, the transmitting stations cannot receive each other nonetheless: LTE radios are still half-duplex, *i.e.* they cannot receive data while transmitting. For safety applications, base stations of a cellular network may even pose unwanted bottlenecks and single points of failure. Thus, only those C-V2X modes not requiring supervision by network infrastructure could fully substitute 802.11p at all. Distributed allocation of resource blocks remains a big challenge in C-V2X, though [129]. Mannoni et al. [180] compare ITS-G5 and C-V2X at broadcasting CAMs in terms of range and latency at varying user densities and bit rates. Their findings show that both technologies have slight advantages over the other depending on the particular conditions. While C-V2X can cover longer ranges because of its more advanced turbo-coding scheme, ITS-G5's PER deteriorates less with increasing node density. The break-even point where ITS-G5 shows a better PER than C-V2X is reached at 150 vehicles/km². Chapters 4 to 5 will show how to counteract ITS-G5's weak points and the effect of those measures.

2.4.6 Summary and Conclusion

The 'big three' (TRC, TPC, and TDC) have an immediate impact on channel congestion. A station transmitting more rarely, or with reduced range, or occupies the channel for a shorter period contributes less to the communication load. Fiddling with other radio parameters such as the size of contention windows or CCA thresholds may mitigate some adverse effects, including channel access while facing many hidden stations. However, their impact is less distinct compared with the 'big three' because they do not control the consumption of channel resources but rather modify the channel access procedure.

ITS applications may actively contribute to lessening channel congestion by keeping their payload lengths short. An already standardised example is the 'low-frequency container' of CAMs, which is only added every 0.5 s.

Omitting this container reduces the payload length by 200 B assuming its history contains 23 path points. Similarly, *TR 103 562* [175] discusses techniques to avoid dissemination of redundant environment objects in CPMs. Since multiple road users may detect an object at the same time, it is not necessary to announce this object in all CPMs. However, deciding which object to omit without hampering the usefulness of collective perception is difficult. Redundancy mitigation is thus only to be applied beyond a – yet undetermined – network channel load [175]. In the further course of this thesis, optimising the payload length is not a goal but left to applications' developers. Instead, mechanisms supporting developers to make most out of IVC in general are of interest.

Prior work has already shown the benefits when congestion control is also involved close to services. Reduced queuing delays and the possibility to orchestrate scarce resources fairly among a growing number of services are a big win. Current approaches [198, 147, 179] consider heterogeneous service demands to some degree, but have difficulties with irregular transmission schemes. Considering IVC as part of a cyber-physical system, services managing platoons and alike would benefit from an indication of how short- to mid-term communication constraints evolve so they can adapt their physical processes in time. Beyond resource orchestration, contracts between services and congestion control may help to avoid race conditions within stations and support services in scheduling their transmissions and planning payloads. Especially packet forwarding is ignored by congestion control entirely so far, while regular beaconing is well studied. A consistent DCC design across the ITS-G5 stack taking each layer's requirements into account is missing, though.

2.5 SIMULATION TOOLS

Network simulations are the logical choice for evaluating VANET's QoS as conducting large-scale field tests is not tractable. Before introducing Artery in Section 3.2, the V2X simulation framework developed as part of this thesis, related simulation tools are presented in the following. Only simulation tools whose sources are accessible are considered for two reasons: First, modifications to the standardised ITS-G5 stack are proposed in follow-up chapters which need to get realised in the simulation model. Second, open source simulation environments lower the barrier for other researchers to reproduce the simulation results as the costs to replicate the setup are negligible.

Two major network simulation ecosystems exist with OMNeT++ and ns-3. Both are discrete event simulators, *i.e.* the state transitions in the models occur only when a respective event is scheduled. These events have no duration itself but only a time point at which they happen according to the model. Consequently, the simulation core advances the simulation time only from one event to the next and no computation is needed between events. As long as the computing operations associated with state changes triggered by events are shorter than the gap between events, the simulation can run faster than real-time.

ns-3 ecosystem

The ns-3 simulator is the current incarnation of an open-source network simulator family. A large and vivid community uses ns-3 for numerous network simulation tasks. This huge impact of ns-3 and its predecessors on the simulation of computer networks won it the Association for Computing Machinery (*ACM*) SIGCOMM Networking Systems Award in 2020 [5].

Out of the box, ns-3 ships with models to simulate IEEE 802.11 radios as used in VANETs. However, the support for full-fledged ITS-G5 stacks, especially with respect to upper layers as GN and DCC, remains limited. A notable effort to extend ns-3's capabilities in the ITS domain has been the iTETRIS project [92]. The iTETRIS platform allows running co-simulations of ns-3 for the wireless communication along with SUMO for the simulation of vehicles' mobility. Beyond this simulator coupling, iTETRIS contributes an implementation of the ITS-G5 protocols to ns-3. Though Rondinone et al. [92] claim that iTETRIS is 'standard compliant', the lack of test cases makes it hard to verify how closely it follows the ETSI specifications of today. The review of iTETRIS' sources reveals that its GBC implementation does not support CBF at all. Furthermore, DCC is completely missing except for basic CBR monitoring. Unfortunately, development and maintenance of the iTETRIS code base have ceased since the initiating projects have ended. The most up-to-date version has been found at EURECOM with the latest changes dating back to 2018 [6].

Just like iTETRIS, Kühlmorgen et al. [149] also use ns-3 and SUMO to simulate ITS-G5 networks. They claim to use 'full protocol stacks', however, have not published any sources of their ITS-G5 simulation model. At least their code to couple ns-3 with SUMO has been made available, though [7].

5 https://www.sigcomm.org/content/sigcomm-networking-systems-award, accessed on 19th May 2021
6 https://gitlab.eurecom.fr/iTETRIS/iTETRIS-release, accessed on 19th May 2021
7 https://github.com/vodafone-chair/ns3-sumo-coupling, accessed on 19th May 2021

OMNeT++ ecosystem

OMNeT++ on its own is just a discrete event simulation kernel and accompanying tools for development and data analysis. Hence, further frameworks providing the actual simulation models are always required with OMNeT++. The INET framework is a popular and actively maintained suite of models for various kinds of computer networks. In particular, INET includes models for 802.11 radio devices, which can be tuned according to ITS-G5 specification. Though some MANET routing protocols are available in INET, it does not ship with any VANET protocols. In summary, INET is capable to simulate the lower layers of an ITS-G5 station but lacks all features belonging to GN and layers upon GN.

When it comes to simulation of VANETss with OMNeT++, the Veins framework [69] is an extremely popular choice. Veins brought the coupling of OMNeT++ and SUMO; the availability of realistic vehicle movement has since become is prerequisite for state-of-the-art VANET simulations. However, Veins solely realises the US variant of VANETs, *i.e.* it centers around use cases that can be realised with BSMs. As a matter of fact, the ITS-G5 model developed as part of this thesis started as an extension for the Veins framework to overcome this shortcoming.

In 2020 the MOSAIC co-simulation framework for connected and automated mobility became available under the umbrella of the Eclipse foundation[8]. The centrepiece of MOSAIC is its runtime infrastructure, which allows to couple many simulators as federates. Among others, MOSAIC features adapters to run ns-3, OMNeT++ and SUMO in simulation federation. While its built-in 'Simple Network Simulator' is advertised to support GBC, the actually implemented forwarding algorithms are much simpler than the standardised algorithms of GN. Hence, MOSAIC's network simulation model is ultimately just as elaborated as the model of the employed federates, *e.g.* ns-3 or OMNeT++.

All in all, open source simulation models of the ITS-G5 system are a rare specimen. Those publicly available are either unmaintained or lack aspects that are essential for QoS evaluations, *e.g.* DCC and correctly behaving GN. As a remedy of this unsatisfactory state-of-the-art, a novel simulation model suitable for QoS simulations of ITS-G5 VANETs is outlined in Section 3.2.

2.6 SUMMARY

As outlined in this chapter, the communication in VANETs is a different breed compared to universally known IP networks. This circumstance reflects in the standardisation and message dissemination patterns: While IP traffic is

8 https://www.eclipse.org/mosaic/, accessed on 19th May 2021

usually conceived as continuous data flow, ITS messages can be interpreted standalone, *i.e.* without knowing prior messages. Also, most ITS packets are not destined to a single entity but to groups of stations, which collaboratively forward packets in these constantly in terms of connectivity changing VANETs.

Despite these fundamental differences between IP networks and VANETs, wireless IP and ITS-G5 networks share a common radio technology with 802.11. Consequently, both network types possess the priority levels offered by EDCA and the CSMA/CA channel access scheme. These commonalities are helpful regarding the availability of adequate simulation models: After some minor parameter adjustments, ITS-G5 radios can be easily simulated with state-of-the-art network simulators such as ns-3 or OMNeT++. However, ITS-G5 specific layers are only poorly covered in the readily available simulators.

Controlling interference and avoiding saturation of the radio channel is essential to achieve good performance in wireless networks, no matter if IP or ITS-G5 protocols are used. Congestion control is handled uniquely by the DCC cross-layer of ITS-G5 to pay tribute to ITS peculiarities such as the considerably more dynamic network topologies. Due to DCC's nature as a cross-layer, *i.e.* an entity which interfaces with several other layers, its impact on the overall VANET performance has to be studied holistically. Otherwise, neglected side effects and feedback by applications, the dissemination process, and other network participants might falsify the assessment.

3 METHODOLOGY FOR EVALUATING QOS IN VANETS

A profound evaluation of QoS mechanisms designed for VANETs requires consideration of several aspects. First of all, QoS needs to get evaluated in scenarios exhibiting the typical characteristics of vehicular communication. An assorted set of scenarios is worked out in Section 3.1. These include varying speed patterns – high speed on motorways versus low speed in city traffic – and channel characteristics influenced by buildings and surrounding traffic. The scenarios also comprise sets of applications and their typical communication behaviour, such as triggering of certain messages depending on traffic context and vehicle dynamics. Section 3.2 deals then with modelling essential VANET components for simulation. This effort comprises the characteristics of ITS-G5 radios as well as protocol behaviour, such as GN affecting channel usage. Finally, appropriate QoS metrics are identified in Section 3.3 and discussed.

3.1 SCENARIOS OF VEHICULAR COMMUNICATION

Analysis of VANET communication depends highly on the employed road topology and surroundings. Foremost, all vehicles are bound to roads and thus cannot move around freely in space. Speed limits, traffic lights and right of way further constrain the vehicle mobility. Physical vehicle properties such as its engine power and brakes limit the achievable acceleration and deceleration.

3.1.1 Traffic Simulation and Driver Behaviour

SUMO has become the de-facto standard in academia for traffic simulation with over 35.000 downloads every year [137]. Two reasons support its popularity and suitability for wireless communications: First, it is freely available as open-source software with active core developers from German Aerospace Center (*DLR*) and a vivid community. Second, its genuine Traffic Command Interface (*TraCI*) eases integration with other software components.

SUMO is called a microscopic traffic simulation, *i.e.* individual vehicles are simulated each with its individual characteristics and driver behaviour. In contrast, macroscopic simulations operate at a coarser level of detail, such as vehicle density on road segments. If simulated vehicles are assembled from individual components, this level of detail is referred to a sub-microscopic.

The superiority of microscopic traffic models over simple waypoint models or static traces has been discussed earlier by Dressler et al. [59]. Since every vehicle equipped with IVC technology is identifiable as a unique entity in a VANET, at least a microscopic traffic simulation is mandatory. Detailed models of vehicle components are usually not necessary, because information of this granularity is not encoded in V2X messages. For example, the engine running speed is not of interest from the viewpoint of communication but the vehicle's speed. Position, speed, and heading can be retrieved from SUMO for each vehicle without further ado.

As outlined in Section 3.2.5, missing vehicle details can be modelled in the network simulation as well. While in SUMO vehicles do not possess environmental sensors, Artery augments vehicles by its environment model with individual sensors such as radar and cameras to perceive neighbouring objects. Still, this environment model builds upon basic data provided by SUMO: Outlines of perceivable objects are determined by position, heading and dimensions of SUMO vehicles.

With respect to vehicle mobility, relying on a microscopic traffic simulation such as SUMO assures realistic movement patterns for VANET nodes. Vehicles do not travel randomly but on given roads and respect the traffic rules, *e.g.* right of way, one-way streets, as well as speed limits. Yet, vehicles do not behave uniformly. Both the vehicle class and individual drivers affect the observable vehicle movement. Lorries accelerate more ponderous due their heavy masses and need larger gaps for lane changes. Likewise, human preferences are incorporated in terms of eagerness for changing lanes, length of safety gaps and deviating obedience to speed limits. SUMO allows to adjust those behaviours via the employed car-following and lane-changing models' parameters. If parameters deviating from SUMO's default values are used, the particular changes are mentioned in the following.

3.1.2 Roads and their Surroundings

Although each vehicle can be seen as an autonomous entity in a microscopic traffic simulation, decisions taken by the aforementioned driver behaviour models are not isolated but depend on surrounding vehicles in most cases. Hence, the modelled volume of traffic, as well as traffic flows from sources

to destinations, affect vehicle mobility. Such traffic demand is statically defined in contrast to the dynamic vehicle behaviour in the previous section. Moreover, traffic demand is closely linked to a particular road topology.

Road topologies can be either be generated synthetically or imported from actual maps. SUMO comes with tools to build road networks based on OpenStreetMap (*OSM*) data [207] conveniently. The advantage of popular synthetic road networks such as grids is the ease to resemble them in less advanced simulations and thus increase the degree of comparability between simulation results. Obviously, road networks based on map data reflect actual courses of roads more realistically.

The vast amount of possible road networks requires focussing on a few distinctive scenarios. These scenarios shall offer an apt context for V2X services. Use cases may be defined only for certain road types, e.g. DENMs indicating a traffic jam shall only be triggered in non-urban environments according to C2C-CC [169]. Additionally, the selected set of road networks shall comprise the following categories, representing the majority of traffic environments.

Motorways

Motorways allow the highest speeds among all road types. Some V2X services such as CAM are triggered by position differences and thus tend to generate messages at a high rate on motorways. Since at least two lanes exist for each driving direction, vehicles can be densely packed, especially in case of traffic jams. Absence of buildings enables unobstructed propagation of radio waves for the most part. Attenuation by other vehicles is the prevailing factor on signal path loss.

Urban Environment

Closed line of houses in urban environments restricts the LOS usually to a single street. At intersections, this may lead to sudden rises in received packets as cross-traffic comes into sight. The average speed in cities is comparatively low and cannot legally exceed 50 km/h in most countries. Yet, stop-and-go traffic and regulated intersections cause recurring acceleration and deceleration phases. Intelligent traffic lights can also emit specific V2X messages and thus induce additional packet load at intersections.

Rural Environment

If present, vegetation such as forests limits the transmission range predominantly in rural environments. In Germany, speeds up to 100 km/h are common on rural roads. Rural environments can feature urban characterist-

ics at small scale, *e.g.* at the heart of villages. Shadowing can vary rapidly between LOS and NLOS because of detached houses. Congested traffic is uncommon, however.

Mixed Environment

With growing map sizes, it becomes more unlikely that a map represents only one category of environment. The transitional regions of mixed environments can be used to evaluate the adaptation capability of V2X communication. Changing the presence of nearby obstacles, for example, directly affects the transmission range.

The transition from one environment category to another can be more or less distinct, *e.g.* whether a city border to its surroundings is sharp or both blend into each other. Also, environments may overlay such as motorways crossing city areas.

3.1.3 A Set of Representative Traffic Scenarios

Although simulations are far easier to conduct than field tests, execution time forbids to run simulations with every possible permutation of parameter settings. This also applies to traffic scenarios, and thus this section carefully selects a small set of scenarios which represents the most typical environments. The presented catalogue of maps is the foundation for the subsequent evaluations of VANET communication.

LuST Scenario for Urban and Motorway Traffic

In general, reuse of existing scenarios is preferred in favour of better comparison between publications. This is especially so if these maps have been used by other publications already. Hence, the renowned LuST scenario [127] is a good candidate, which has become popular in the ITS community with 85 citations counted by Google Scholar as of 7th December 2020. This scenario primarily covers Luxembourg city plus some of its outskirts with almost 156 km² and 929.5 km of roads in total. The inner-city area is, except for the north side, encircled by a three-lane motorway, which has a speed limit of 130 km/h. Thanks to the overall size of this scenario, it features not only densely built-up areas but also sub-urban residential districts intermixed with arterial roads.

Special care has been taken by Codeca, Frank and Engel [127] at modelling the traffic realistically, based on data gathered by the municipality and floating car data. Consequently, this scenario features mixed traffic with various kinds of passenger cars, lorries and buses over a period of 24 hours. By shifting the observed time window, one can vary the traffic density from a few hundred

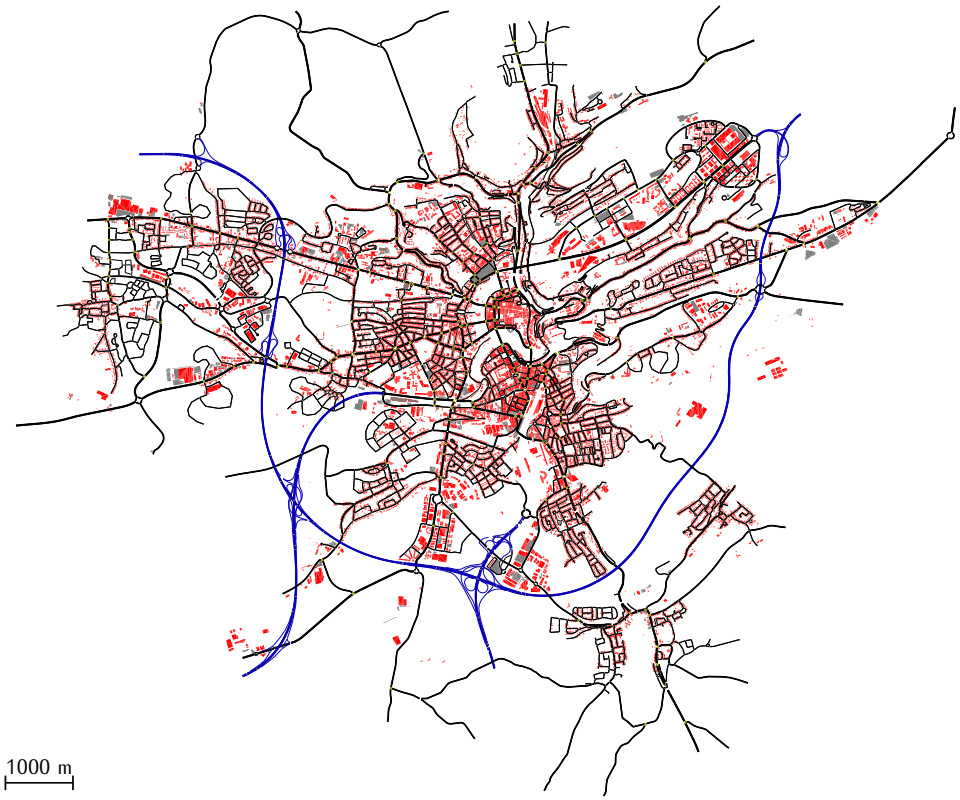


Figure 3.1: Screenshot of LuST scenario: Red polygons are buildings, grey polygons are parking facilities; Blue roads are motorways, other roads are black.

vehicles at night to over 4500 concurrent vehicles during rush hours. With respect to traffic demand, the scenario ships with four variants of generated mobility traces:

- Routes calculated by Dijkstra's shortest-path algorithm without congestion (Dynamic User Assignment) or
- or the iteratively with congestion (Dynamic User Equilibrium) and
- with *static* traffic light logics or
- *actuated* traffic lights by loop detectors

Codeca, Frank and Engel suggest using the *actuated* traffic lights as these result in 'efficient, realistic, and easy-to-use scenario'. Execution of the *Dynamic User Assignment* variant takes about 1.5 times as long as *Dynamic User Equilibrium*, because the former relies heavily on dynamic re-routing of vehicles during simulation. However, experiments have shown that the *Dynamic User Assignment* variant is able to maintain more running vehicles in the simulation across various SUMO versions. Since more nodes competing for channel resources are favourable when evaluating VANET performance, simulation experiments in this thesis employ *Dynamic User Assignment* with actuated traffic lights as the default option. For reference, with LuST 2.0

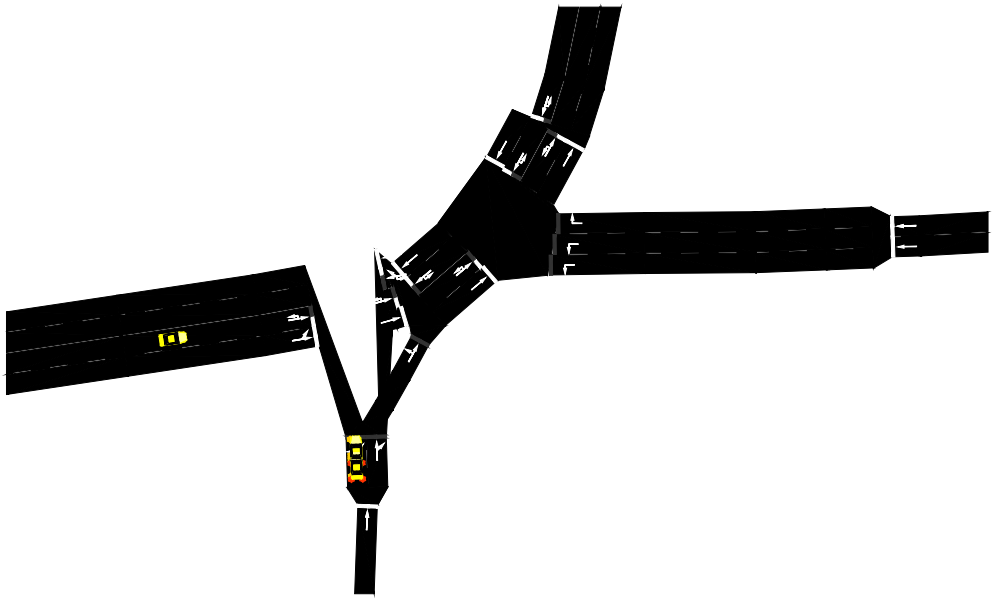


Figure 3.2: Typical map error found at intersections in TAPASCologne (Rendering by SUMO 1.3.1)

and SUMO 1.6.0, 4771 vehicles are on the streets at 08:00 morning with *dua.static.sumocfg* versus 4862 vehicles with *dua.actuated.sumocfg*. As a bonus, the LuST scenario bundles polygons of all buildings. One can incorporate the outline of these buildings in radio propagation models, such as GEMV² or the obstacle shadowing model used by Veins [68]. However, no vegetation is included though this is only a minor issue in urban environments.

Two urban alternatives to the LuST scenario were considered but ultimately discarded: Codeca and Härri have also created a SUMO scenario of Monaco, called *MoST* [140]. This scenario puts emphasis on elevations and thus a 3D road network in contrast to the flat LuST scenario. Unfortunately, the availability of VANET radio models tuned for mountainous environments including tunnels is lacking behind. While MoST is a well-crafted scenario on its own, this thesis sticks with LuST to avoid apparent imprecisions that would be introduced by other models when combined with MoST. SUMO's scenario repository includes TAPASCologne[1], a map covering an area of roughly $28 \times 32\,\mathrm{km}^2$. While the sheer size of this scenario is impressive, many of its intersections are broken, as shown in Fig. 3.2. During the simulation of TAPASCologne, the number of concurrent vehicles can easily exceed 32.000. This amount would create a massive computational burden for the wireless communications simulation. The advantage of simulating areas several times larger than the most optimistic transmission ranges is also unclear.

1 https://sumo.dlr.de/docs/Data/Scenarios/TAPASCologne.html, accessed on 12th November 2019

DekiNet2 for Rural and Motorway Traffic

Unfortunately, no established scenario representing rural environments has been found. Traffic engineering, the original domain of SUMO, may be more interested in cities where infrastructure is planned and optimised at times. Bai, Stancil and Krishnan [56] discovered in their field measurements that rural environments expose communication to particularly harsh channel conditions, though. Thus, an import from OSM data is detailed in the following, to remedy the lack of a rural scenario. By sharing the crafted scenario with the ITS community, further usage of this scenario named 'DekiNet2' is encouraged.

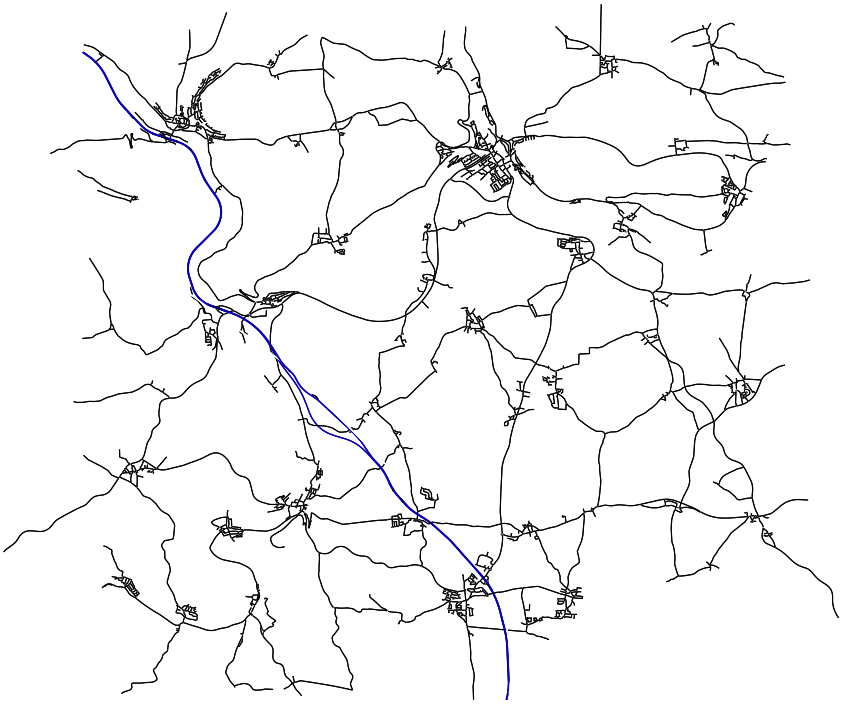
DekiNet2 is the reincarnation of a scenario first used in an earlier paper with focus on a rural environment [9]. Since then, the SUMO toolchain has improved considerably, with `netedit` leading the way. This tool facilitates the process of manually correcting import errors such as broken intersections and erroneous turning options. Furthermore, for DekiNet2 the vegetation has been carefully incorporated to make it usable with advanced radio propagation models such as GEMV².

Based on the OSM dataset from 24th October 2019, the area between 11.222 049° to 11.604 653° East and 48.870 896° to 49.083 973° North has been imported with SUMO's `netconvert` tool. This area includes the two eponymous places ***De****nkendorf* and ***Ki****pfenberg*, quite in the middle of Bavaria, Germany. In numbers, the imported map has a size of 400 km² with a road network of 1158 km length.

Vegetation with foliage is described by polygons in the employed OSM dataset as *natural.wood*, *landuse.forest*, *landuse.orchard*, *landuse.plant_nursery*, *natural.tree*, *natural.tree_row*, and *natural.scrub*. Typical key-value pairs used by the underlying OSM map [208] to describe the environment, including the vegetation, are listed in Table 3.1. In the first step, similar polygon types are relabelled with common names. Areas continuously covered by trees are tagged as *forest* (*natural.wood* and *landuse.forest*), *natural.scrub* as *scrub* and the remaining types default to *tree*. While foliage is treated uniformly at present state in GEMV², those three groups enable the adaptation of attenuation coefficients based on the density of vegetation.

A look at the original dataset revealed that some large vegetation polygons might be overlayed by 'islands' of another polygon type. If vegetation polygons are imported naively into the foliage model, such overlapping geographical polygons introduce errors. For example, a particular place cannot be forest and farmland though one overlays the other. Hence, a post-processing algorithm has been constructed to edit polygons without overlapping.

(a) With polygons representing physical geography, towns and buildings



(b) Only roads, motorway A9 in blue

Figure 3.3: Overview of DekiNet2 map

(a) Original polygons **(b)** Merged polygons

Figure 3.4: Detail of LuST scenario with buildings (red)

SUMO's XML format for storing polygons induces another constraint: It can only represent polygons without holes. Such holes are created during post-processing by resolving cases where one polygon type is completely within another. In general, the post-processing algorithm stores polygons as a sequence of coordinates describing its exterior and 0 to n interior (non-overlapping) rings. Before writing polygons back to the target XML format, polygons with interior rings are normalised by cutting them in two halves through the centroid of the first interior ring. This step is repeated for each half recursively until no interior ring remains, *i.e.* at most n times.

Polygons of key *landuse* with assigned value of either *commercial*, *industrial*, *residential*, or *retail* are unified as *town* polygons. These polygons have no relevance for radio propagation; however, one can determine if a position lies in a town. Such differentiation is handy as some IVC use cases are specifically designed for urban or non-urban environments and shall only be operational in those areas.

Most complexes of buildings are represented as multiple, adjacent polygons in the original data set. For example, each house with a dedicated house number in terraced housing is represented by a separate polygon. However, from the perspective of radio propagation, those individual, adjacent houses are a single, massive obstacle. Radio propagation models, that calculate their signal attenuation based on diffraction at corners, such as GEMV² presented in Section 3.2.1, benefit from merging these buildings into a single polygon. This merging process can also be applied to the building polygons of the LuST scenario, as shown in Fig. 3.4. It has to be noted that inner courtyards are 'filled up' by this process, though. While this has no impact on the GEMV² model, one should stay with the original polygon outlines if static obstacle attenuation is calculated based on crossed distances through buildings.

Table 3.1: Conversion of OSM polygons for DekiNet2

OSM map feature	*polyconvert* type	colour
natural.wood landuse.forest	forest	
landuse.farmland landuse.farmyard landuse.village_green	farm	
natural.grassland landuse.grass landuse.meadow	grass	
waterway natural.water natural.wetland natural.spring landuse.basin landuse.reservoir	water	
natural.sand	sand	
natural.fell natural.bare_rock natural.scree landuse.quarry	rock	
natural.tree natural.tree_row landuse.orchard landuse.plant_nursery	tree	
natural.scrub	scrub	
landuse.commercial landuse.industrial landuse.residential landuse.retail	town	
building	building	
amenity.parking	parking	
*	<discard>	

geographical polygons (ascending order)

Road traffic census published by the Bavarian state[2] has been used to model traffic demand on motorway A9. Counter 70349001 situated between Denkendorf and Kipfenberg has registered 73176 vehicles in 24 hours for both directions in 2015. 10346 of these vehicles belong to heavy goods traffic, *i.e.* 14.1 % of total traffic volume. Based on these numbers, traffic flows with 1520 vehicles per hour are created for each direction. With a probability of 14.1 % a lorry is inserted per flow. Lorries need about 920 s for travelling the complete motorway distance, and the motorway traffic settles up to a steady-state with roughly 640 vehicles after the first 16 min. Cross-country traffic is a mix of 1840 passenger cars and delivery vans with trips visiting three random waypoints. Only trips with minimum a distance of 1500 m are considered. In alignment with the motorway traffic, all cross-country vehicles are inserted by simulation time $t = 920\,s$. This configuration shows a stable number of about 2400 vehicles for several minutes from then.

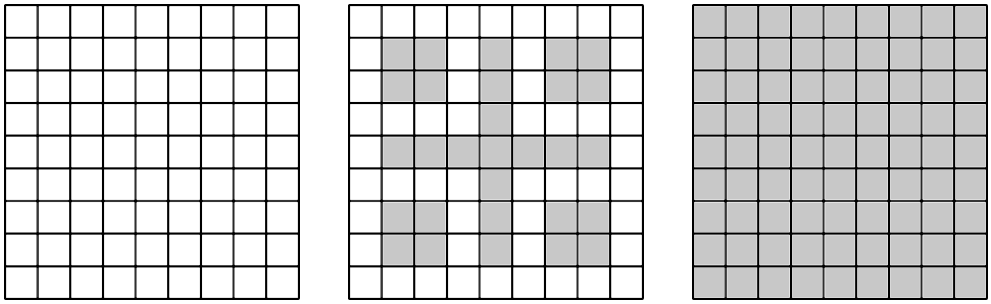
Synthetic Grid

Synthetically generated road topologies do not represent real traffic environments but rather approximate such an environment in an idealised way. A popular synthetic road topology is the grid layout, inspired by the road layout found in Manhattan and Mannheim. The motivation to include such an idealised scenario is manifold: First of all, having simulation results based on a grid scenario facilitates comparability with third-party publications relying on such a scenario primarily. Moreover, other toolchains may struggle at importing real map data but succeed at reproducing synthetic grids with their uniform structure. Following the building block concept, synthetic scenarios can also be varied with less effort. For example, increasing the number of lanes to host high vehicle densities without congestion is trivial.

The particular grid scenario used in the further course will be called 'Griddy', the *grid* roa*d* topolog*y*. 10×10 junctions, each separated at their centres by 200 m, are taken as a basis. Each street edge is assigned a speed limit of 50 km/h. The blocks enclosed by the streets can be occupied by quadratic buildings with an edge length of 180 m. As shown in Fig. 3.5, three occupation variants are proposed, that allow the investigation of shadowing effects by buildings at varying degrees: A plain grid without any buildings, buildings arranged as a plus symbol with four insular blocks at its corners, and buildings at every block.

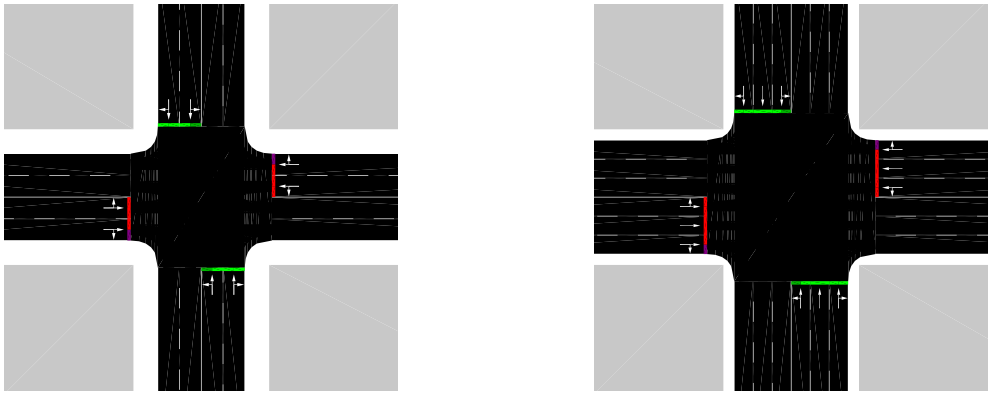
With the separation of junctions and dimension of buildings fixed, the resulting street width is always 20 m. However, the number of lanes can be adopted to the simulated traffic density without changing the shadowing characteristics by surrounding buildings. Figure 3.6 shows an intersection

2 BAYSIS Datenabfrage (Straßenverkehrszählungen) [English: Data query (road traffic census)] https://www.baysis.bayern.de/web/content/verkehrsdaten/SVZ/strassenverkehrszaehlungen.aspx, accessed on 5th December 2019



(a) plain **(b)** plus islands **(c)** full

Figure 3.5: Variants of synthetic grid scenario Griddy



(a) Two lanes **(b)** Three lanes

Figure 3.6: Detail of intersection in Griddy scenario variants

in detail with two or three lanes per direction. Road markings indicate the turning possibilities of each lane at the respective intersections. The variation of lanes and arrangement of buildings thus sum up to six static layouts of Griddy.

Two major flavours with respect to traffic demand can be differentiated: One is based on randomly generated trips, the other one incorporates straight routes from one border edge to the opposing one.

Traffic demand with random trips uses 720 individual flows, *i.e.* vehicles belonging to the same flow share common departure and arrival edges. Furthermore, eight randomly selected waypoints appertain to each flow, which elongates travelling distances of vehicles considerably. Thus, vehicles stay in the network for several minutes compared to the rather short trips when travelling directly from source to destination edge. A destination edge is always an edge at the border of the network, while sources and waypoints can be any edge. Each flow is assigned a probability describing the likelihood a vehicle departs within one second. Flows generate vehicles during the first 60 s. Hence, this time span T, the number of flows F, and their (equal) probability P defines the expectancy value of total vehicles $N = T * P * F$.

Depending on the initial seed for SUMO's random number generator, one can expect around 2160 running vehicles in Griddy network if a probability of 0.05 is used.

Given by the grid layout of 10×10 junctions, 40 straight routes exist from north to south, west to east and respective opposite directions. Again flows are employed to occupy these routes; however, vehicle occurrences are defined as vehicles per hour instead of probabilities. When the predefined target number of vehicles for the network is reached, new vehicles are not directly placed on their start edges but kept in a backlog. This leads to a rather uniform vehicle distribution across the network. The lack of turning behaviours also reduces the risk of traffic jams enormously, *i.e.* with this traffic demand variant more concurrently running vehicles can be added to the network compared to the random trips variant. Observation showed that vehicles need roughly 4 min to reach their respective destinations. Consequently, simulation of wireless communications should skip the first minutes until a steady-state in terms of traffic flow is reached.

The vehicle class distribution is copied from the LuST scenario except for buses. Thus, vehicle behaviour concerning positive and negative acceleration capabilities and speed deviations is mostly identical. In addition to the original vehicle distribution, those classes are not only assigned vehicle lengths but also non-default width and heights. Those vehicle dimensions are used by some geometry-based radio propagation models. Furthermore, SUMO's sub-lane model is enabled to avoid perfect lateral alignment among vehicles. Considering the regular road layout, some lateral displacement is favourable especially when vehicles are considered as radio propagation obstacles.

3.2 SIMULATION MODEL OF VANET COMMUNICATION

With field-tests being prohibitively expensive and hard to coordinate when scaled beyond a handful of vehicles, creating abstracted simulation models is the preferred way to evaluate communication networks. Notably, discrete event simulations are a de-facto standard when it comes to complex network setups. Analytical approaches are not feasible with a sufficient level of detail. For example, Luan et al. design a VANET with rather simple multi-hop forwarding capabilities as a queuing model, which already needs to be solved numerically [150]. More heterogeneous mobility patterns or message generation rules are not within reach, though.

In conjunction with this thesis, a lot of time has been invested in the development of Artery, a sophisticated simulation framework for V2X communication. Figure 3.7 shows Artery's the high-level architecture featuring its

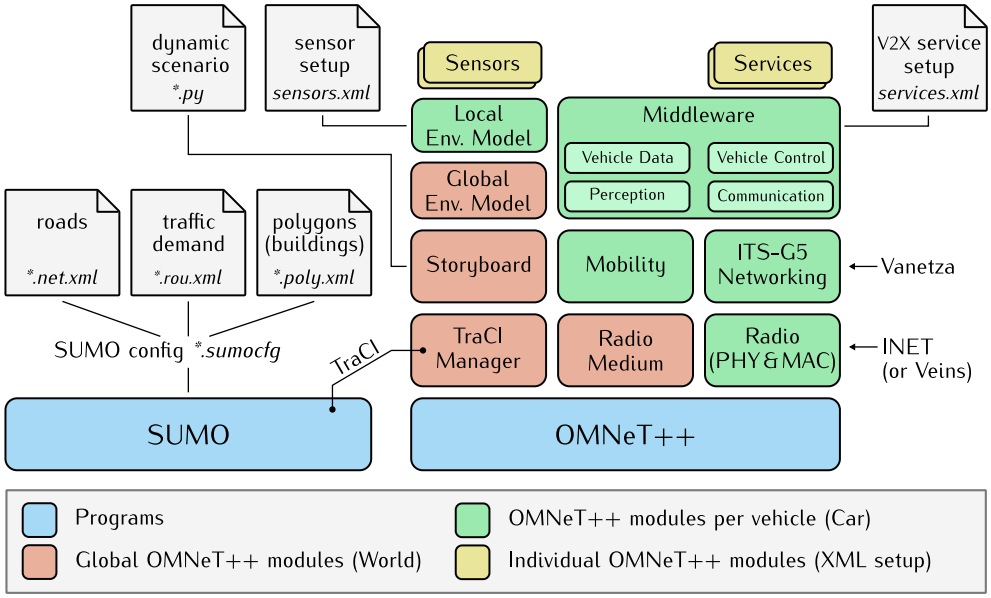


Figure 3.7: High-level architecture of Artery

major building blocks. Development of Artery has been a collaborative effort with Hendrik-Jörn Günther from the beginning. From its humble beginnings as an extension of the Veins framework [69] to support ITS-G5 applications, it has since grown to an independent framework with many distinct features [3]. A cornerstone is the so-called *Middleware* module, which is a vehicle's hub for gathering data for message generation and triggering reactions upon received messages. Further feature highlights are:

- Scriptable scenarios allowing to realise evolving, dynamic scenarios, *e.g.* changing weather conditions (*Storyboard*) [11]
- Vehicle sensors with LOS occlusions checks with respect to vehicles and buildings (*Environment Model*) [130, 15]
- LTE communication as an option beside ITS-G5 radios [10]
- Integration of Software-Defined Radio (*SDR*) hardware for Hardware-in-the-Loop testing setups [14]

Inspired by Veins, simulation of network traffic builds upon the discrete event simulator OMNeT++ whereas the road traffic simulator SUMO provides vehicles' realistic mobility patterns. Both are bidirectionally coupled, *i.e.* network applications can influence vehicle behaviour and vice versa IVC radios are mounted on SUMO vehicles. Whereas Veins re-implements the TraCI protocol, Artery ships SUMO's upstream C++ Application Programming Interface (*API*) and thus supports all SUMO features.

Artery embeds with Vanetza another software project for the ITS-G5 specific protocols. This project has also been developed for this thesis and implements the ITS-G5 specifications as C++ libraries. Vanetza will be discussed in more details in Section 3.2.4.

3.2.1 Radio Propagation

Whether a transmitted signal is ultimately decodable by a receiver is much affected by the signal's strength. The transmitter outputs the signal with a certain power level, and after traversing the radio medium, it reaches the receiver's antenna. Only if the signal is then still strong enough for the receiver's sensitivity, it can be received at all.

None of the models discussed hereinafter considers the relative velocity between sender and receiver. Although one may assume that Doppler shifts hamper packet delivery, Bai, Stancil and Krishnan [56] found no such effect in the data of their field campaign. The distance between sender and receiver as well as the presence of scatterers attenuate signals in the 5.9 GHz predominantly. Similarly, Almeida et al. [189] attest 802.11p communication to be robust against adverse Doppler shifts based on measurements and comparison with simulation results.

Deterministic Propagation Models

As a fundamental propagation model, the Friis free space equation Eq. (13) describes the expected signal loss on an unobstructed path of length d between transmitter and receiver [29]. The received power P_r depends on the transmission power P_t, the antenna gains G_t and G_r, and an attenuation factor $\lambda^2/(4\pi)^2 d^2 L$. This attenuation varies with the signal's wavelength λ and the square of distance d. Other parts of the equation are fixed, including the system loss factor $L \geqslant 1$, which can be used to model cable losses between antenna and amplifier, for example.

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{13}$$

Receptions based on the Friis equation are fast to compute and entirely deterministic. However, it does not take any effects caused by the environment into account. The most prominent effect is caused by the ground all road vehicles are rolling on. It has been shown based on measurements with cars in the 5.9 GHz band, that unobstructed LOS propagation of IVC signals is better described by the full two-ray ground reflection model [78, 50]. By analogy with the Friis Eq. (13), the received power in the two-ray ground reflection model is given by Eq. (14). The idea behind this model is shown in Fig. 3.8: Instead of a single signal path liable to free space loss, two interfering phase-shifted paths are considered [29]. One path is the direct LOS between the two elevated antennas; the other path is reflected once by the ground. Their respective lengths are d_{los} and d_{ref} and their phase difference ϕ is given by Eq. (15). The height of the transmitting antenna

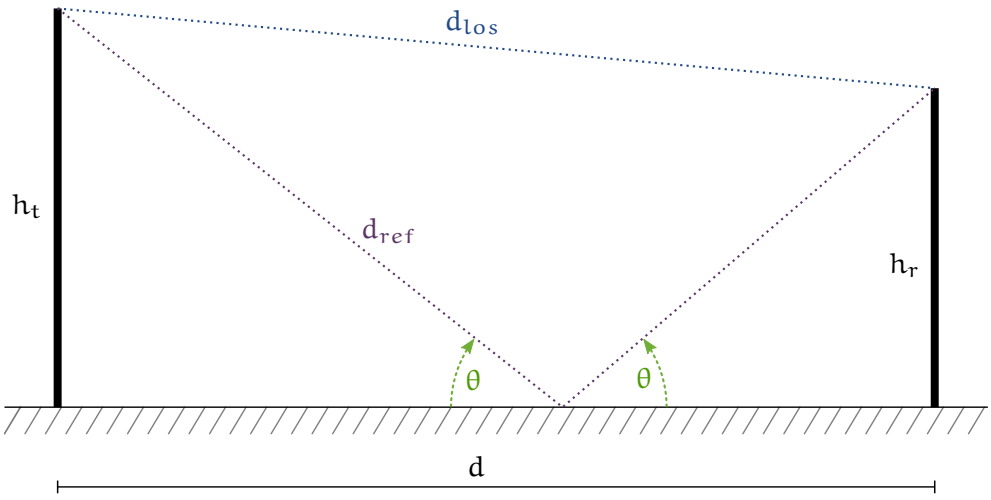


Figure 3.8: Schematic representation of the two-ray ground model

is given by h_t. The receiving antenna is elevated by h_r. Γ is the reflection coefficient, depending on the relative permittivity ϵ_r (a material constant) and incidence angle θ.

$$P_r = P_t G_t G_r \frac{(1 + \Gamma\cos(\phi))^2 + (\Gamma\sin(\phi))^2}{16\pi^2(\frac{d}{\lambda})^2} \tag{14}$$

$$\phi = 2\pi\frac{d_{ref} - d_{los}}{\lambda} \tag{15}$$

$$\Gamma = \frac{\sin(\theta) - \sqrt{\epsilon_r - \cos^2(\theta)}}{\sin(\theta) + \sqrt{\epsilon_r - \cos^2(\theta)}} \tag{16}$$

For huge distances d, the path difference $d_{ref} - d_{los}$ can be approximated by $2h_t h_r/d$. This in combination with $\sin(\alpha) \approx \alpha$ for small α yields to a simplified two-ray ground equation, which is given by Eq. (17) [29].

$$P_r = P_t G_t G_r \frac{{h_t}^2 {h_r}^2}{d^4} \tag{17}$$

However, Sommer, Jörer and Dressler stress the importance to employ the non-simplified model despite its more complex computation to resemble the measured signal strengths in a vehicular setup [78]. Likewise, Rappaport states that the simplified Eq. (17) is only applicable for distances $d > \frac{20\pi h_t h_r}{3\lambda}$. In a typical VANET with $h_t = h_r = 1.5\,m$ this constraint would only be fulfilled by distances beyond ~1 km, which is unrealistic. Consistent with Sommer, Jörer and Dressler [78], the term 'two-ray interference' is used in this thesis to point out that the non-simplified two-ray ground reflection model is used.

One of the major benefits of deterministic propagation models is the good comparability of results from different simulation toolchains. Random effects hampering the PER can be excluded reliably if other parameters are kept equal among them. Furthermore, for some application metrics such as coverage ratios, it is beneficial to know a fixed transmission range a priori. For example, a metric like 'awareness ratio' [6] would be difficult to calculate if the transmission range for a given transmission power would change.

Probabilistic Nakagami Propagation Model

Already in the early days of 802.11 based VANETs, Torrent-Moreno, Jiang and Hartenstein favoured a stochastic Nakagami fading model over the deterministic two-ray ground model [36]. They also point out that the parameters controlling the Nakagami distribution have to be selected according to the communication distance. Cheng et al. derive a Nakagami path loss model based on own measurements in the 5.9 GHz band, specifically for IVC [43]. Nakagami distributions are controlled by two parameters, where m controls its shape and Ω its spread [17]. Higher values of m give the Nakagami probability density function a steeper and narrower shape and thus can represent communication links where fading is very similar, *e.g.* LOS at a close distance. Ω, on the other hand, is used to incorporate the estimated average power at the receiver side. This average power $P(d)$ depends only on the distance d from the transmitter and is calculated using a classic dual-slope model as shown in Eq. (18). It can be noted that for the first case (distances below critical distance d_c) the path loss is identical to the free space path loss model with loss exponent γ_1. With $\gamma_1 = 2$ and $\gamma_2 = 4$, the power $P(d)$ falls by the inverse-squared distance up to d_c, and even inverse power of 4 for longer distances. This parameterisation is in line with observations noted by Taliwal et al. [35]. While Taliwal et al. suggest switching between the two modes of the dual-slope model at a critical distance $d_c = 160\,m$, Cheng et al. decided to use $d_c = 100\,m$. The more often cited parameterisation by Cheng et al. is used by this thesis in the hopes to facilitate comparison with other works.

$$P(d) = \begin{cases} P(d_0) - 10\gamma_1 \log_{10}\left(\frac{d}{d_0}\right) + X_{\sigma_1} & \text{if } d_0 \leqslant d \leqslant d_c \\ P(d_0) - 10\gamma_1 \log_{10}\left(\frac{d_c}{d_0}\right) & \\ \qquad\qquad - 10\gamma_2 \log_{10}\left(\frac{d}{d_c}\right) + X_{\sigma_2} & \text{if } d > d_c \\ P(d_0) & \text{if } d < d_0 \end{cases} \tag{18}$$

Table 3.2: Nakagami shape factor depending on distance. Values taken from [43].

Distance					Shape factor m
0.0 m	$\leqslant$	d	$<$	4.7 m	3.01
4.7 m	$\leqslant$	d	$<$	11.7 m	1.18
11.7 m	$\leqslant$	d	$<$	28.9 m	1.94
28.9 m	$\leqslant$	d	$<$	71.6 m	1.86
71.6 m	$\leqslant$	d	$<$	177.3 m	0.45
177.3 m	$\leqslant$	d			0.32

Similarly, Taliwal et al. suggest the Nakagami shape factor to be in the range between 1 and 4 for unobstructed traffic and between 0.5 and 1.0 on highways. These values match the shape parameters given in [43, Table III and IV]. X_{σ_1} and X_{σ_2} are Gaussian distributed noise portions with zero-mean and standard deviations of $\sigma_1 = 5.6\,\mathrm{dB}$ and $\sigma_2 = 8.1\,\mathrm{dB}$ respectively.

Some corner cases have not been mentioned by Cheng et al. explicitly. For clarity, the previously missing third case for distances below the reference distance d_0 has been added to Eq. (18). Also, no particular value for d_0 has been given. Considering the wavelength $\lambda = {}^{c}/_{5.9\,\mathrm{GHz}} \approx 5\,\mathrm{cm}$ a reference distance of $d_0 = 1\,\mathrm{m}$ is definitely in the far-field of the antenna but still so close that free-space path loss is the predominant loss. $P(d_0) = P_{tx} - PL(d_0)$ can be determined by simple free space path loss $PL(d_0) = \frac{4\pi d_0}{\lambda}^2$.

Since the parameterisation of stochastic path loss models is always subject to their fitting with a particular data set of measurement samples, it is hard to determine one set of parameters suitable for all simulations. However, within this thesis, the data set 2 from [43] is selected as it also used by Sjöberg, Uhlemann and Ström [67]. Data set 2 uses lower shape factors compared to data set 1, and thus, the Nakagami distribution is spread wider. Of course, scenarios similar to the measurement environment (sub-urban area with two lanes) will be approximated best by these parameters. Still, adapting the shape factor based on distance is more convincing than using a fixed factor for all communication links because the likelihood for obstructions grows with the link distance. This general effect is well covered by the selected parameters, as shown in Table 3.2.

The authors of the aforementioned Nakagami model themselves motivate to look into another type radio propagation model, though:

> [...] houses and buildings between the vehicles increased the attenuation and intermittently obstructed the line-of-sight. This observation suggests the desirability of a multi-state model, with different states being applicable when a line-of-sight does and does not exist between the vehicles. [43]

While the Nakagami model adapts its distribution parameters based on the transmission distance and can be tuned to a given environment, it has its limitations when predicting the path loss in changing environments, *i.e.* where vehicle densities vary and buildings are spread unevenly.

Geometry-based Propagation Model GEMV²

GEMV² is a multi-state model as it classifies each communication link into four main types: LOS, Non-Line-of-Sight due to obstructions by vehicles (*NLOSv*), Non-Line-of-Sight due to obstructions by buildings (*NLOSb*), and Non-Line-of-Sight due to obstructions by foliage (*NLOSf*). Based on this classification, one particular sub-model is selected that encapsulates the applicable path loss algorithm in each case. These algorithms can be entirely deterministic, stochastic or a mixture of both. GEMV² has been created by Boban as part of his PhD thesis [71]. A denser description of GEMV² can be found in his follow-up journal article [33]. This section gives only a brief overview of the main features and some peculiarities concerning the Artery framework. Artery's implementation of GEMV², which is depicted in Fig. 3.9, has been acknowledged by Mate Boban on the GEMV² website[3]. In contrast to the original Matlab implementation, the sub-modules can be easily exchanged via the OMNeT++ configuration. Any path loss model fulfilling INET's `IPathLoss` interface can be used as sub-model, which enables extensive customisations.

The link classification is based on the geometric shapes of static obstacles such as buildings, shapes of trees and other foliage, and outline of vehicle bodies. While former two are loaded once due to their static nature at the begin of simulation, vehicle shapes are updated every SUMO simulation step, *i.e.* whenever vehicles change their position and orientation. All geometric properties are loaded from SUMO through the TraCI connection which reduces the amount of configuration in Artery itself. Intersection tests of the direct line between transmitter and receiver antenna are executed in the following order:

1. static obstacles, *i.e.* mostly buildings
2. foliage
3. vehicles

If a particular type of obstacle intersects with the given line, the accompanying NLOS variant is selected. Only if none of them intersects the link is considered to be of type LOS.

By default, the aforementioned two-ray interference model is used for LOS links. A fast log-distance path loss model is employed as the default option for NLOSb for performance reasons. However, a more accurate variant is

3 http://vehicle2x.net/download, accessed on 31st July 2019

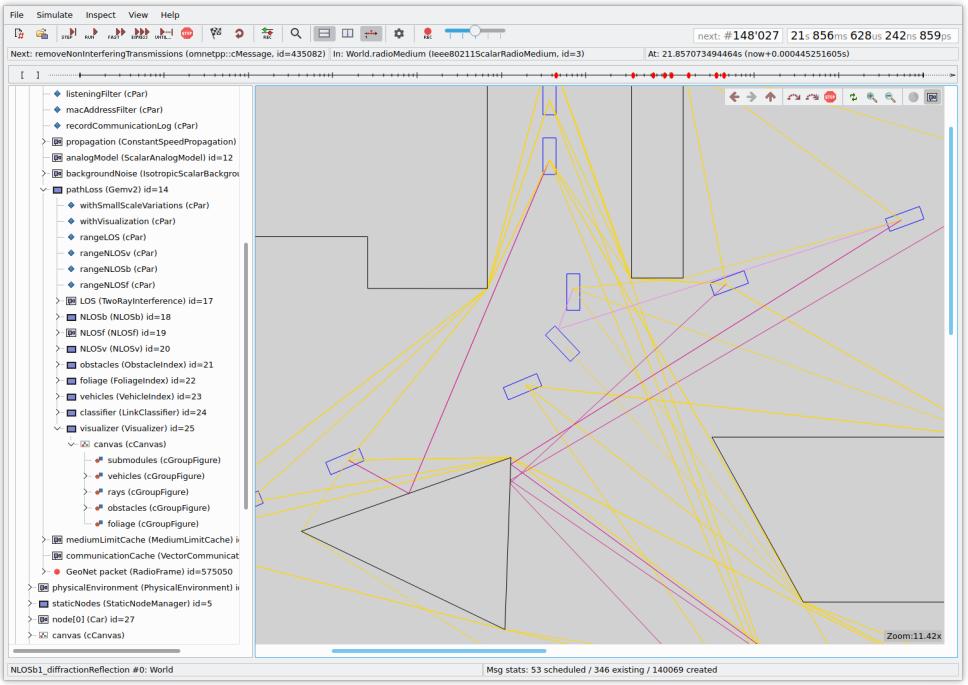


Figure 3.9: Visualisation of GEMV² in Artery with reflection rays (purple) and diffraction rays (yellow). Static obstacles are outlined in black, vehicle shapes in blue.

implemented as well, which takes diffraction at corners of obstacles and single reflections by building and vehicles into account, as shown in Fig. 3.9. When vehicles are the primary obstructions, NLOSv calculates the diffraction paths sideways and above each obstructing vehicle. These calculations take the antenna heights as well as cuboid vehicle shapes into account. Last but not least, foliage attenuates the signal by 2.33 dB per traversed meter.

Aforementioned sub-models build an entirely deterministic path loss model. Optionally, GEMV² can be enhanced by stochastic small-scale variations. These are modelled as normal-distributed attenuations which are adopted dynamically according to link type. Vehicle and building densities within the ellipse that has the transmitter and receiver at its foci are also considered.

Simulations conducted as part of this thesis use the described Nakagami model and GEMV² exclusively. To the best of my knowledge, GEMV² is the only radio propagation model specifically designed for IVC as of today, that can cope with a variety of communication links concurrently. Still, the Nakagami model is a useful addition to cross-check and also considerably easier to implement. Ease of implementation paves the way to verify simulation results obtained with Artery by other toolchains.

As mentioned in Section 2.4, stochastic propagation models as Nakagami accentuate hidden station phenomena more than deterministic models. GEMV² offers a compromise with its classification of link types and de-

terministic inclusion of obstacles where stochastic models can only cover such a wide range of link characteristics with large variances of their random variables. As each single reception decision draws independently from the employed model's random variables, the smaller variances added on top of GEMV² are less likely to overemphasise hidden stations. This behaviour is a compromise between entirely deterministic and purely stochastic models.

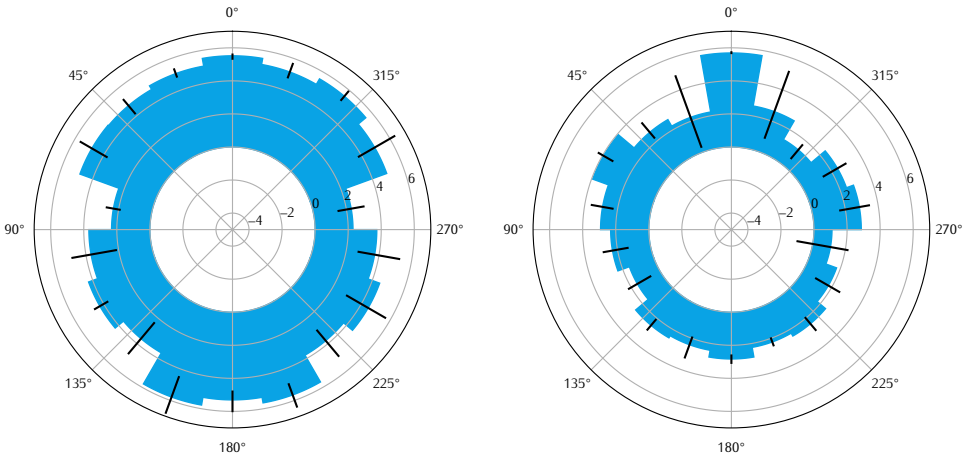
3.2.2 Antenna Patterns

Besides the radio channel's characteristics, the used antennas by transmitter and receiver affect the signal strength. With the Friis Eq. (13), two parameters G_t and G_r have already been introduced, which denote the individual antenna gains. In the simplest case, one can assume antennas with unit gains, *i.e.* isotropic antennas that radiate equally in every direction [29]. Since IVC relies on broadcast communication in the vast majority of cases, it is reasonable to assume that vehicle manufacturers want to deploy antennas with an omnidirectional characteristic and preferably high gain. By omnidirectional characteristic, a uniform radiation in all directions (mostly) parallel to the ground is meant. Since other communication parties are also ground-based or only slightly elevated, *e.g.* RSUs, no reason exists to radiate straight up in the sky or to the ground.

The plethora of vehicle shapes and antenna types makes it impractical to cover all their varieties in simulation. Kwoczek et al. [64] showed by measurement that antenna designs exist achieving at least unit gain (*i.e.* 0 dBi) in driving direction and up to 5 dBi gain perpendicular to driving direction. They also discovered that a panorama glass window could severely reduce the antenna performance in a range of −15 dBi to −20 dBi. However, with this known to be problematic, it can be assumed that such a flawed antenna setup would not get deployed in series production and is thus not of practical relevance.

While Eckhoff, Brummer and Sommer [115] argue that highly directional antenna patterns need to be considered in VANET simulations as it affects the horizon of safety applications, there are good reasons to disagree with this argumentation: No particular direction is necessarily more relevant for safety applications since not only traffic straight ahead but also cross-traffic induces risks. The multitude of IVC use cases thus does not demand for directional antennas but omnidirectional antennas.

Even for motorcycles, where no antenna can be mounted on a roof obviously, and an engine block attenuates the signal of low-mounted antennas, almost omnidirectional antenna gains can be achieved as shown in Fig. 3.10. These figures are based on unpublished measurements commissioned by the



(a) Motorcycle A **(b)** Motorcycle B

Figure 3.10: Antenna gains [dBi] measured for two motorcycles, each with two antennas combined by antenna diversity (Maximum Ratio Combining). Blue segments represent the mean gain in this direction (0° is heading front), the black bars indicate the 10^{th} and 90^{th} percentile.

Connected Motorcycle Consortium. Two antennas have been mounted on both motorcycles, one close to the headlight, the other at the tail. The plots show the antenna gains at 5.905 GHz averaged for segments of 20° width.

Simulations investigating the overall network performance can employ simple, uniform antenna models with a moderate gain, *e.g.* 3 dBi, for two reasons: First, one can assume that omnidirectional antennas are generally favourable to realise common IVC use cases. Second, neither front (see [64]) nor side directions (see Fig. 3.10) necessarily exhibit higher gains in real-world antenna designs. Nevertheless, if a particular vehicle is investigated, one may want to use a tuned antenna model representing this vehicle's directional preference. Measurements indicate that gain differences between directions usually do not exceed 6 dB, however.

3.2.3 Radio Devices compatible with ITS-G5

With a focus on QoS aspects of VANETs, the simulation model of the wireless network interface card has to cover over-the-air packet durations, interferences and channel access behaviour properly. Artery has an abstraction layer to use various implementation by providing `RadioDriver` adapters. In the OMNeT++ ecosystem, the 802.11 model from the INET framework[4] is recommended because of its comprehensiveness, comparatively large user base, and active development. This section highlights necessary parameterisation and custom extensions for modelling VANET traffic. On top, existing features are checked to show the expected behaviour.

4 https://inet.omnetpp.org (Visited on 5th August 2019)

Channel access behaviour and interference

Most aspects related to 802.11 channel access are readily available in the INET framework. Namely, basic CSMA/CA as well as its QoS extension EDCA, can be employed without further ado. Furthermore, the lack of any 802.11 management frames because of the OCB mode can easily be achieved by configuring INET 802.11 stations as ad-hoc nodes. However, one needs to take care to configure EDCA correctly for VANET communication because the length of the AIFS per AC differs from usual WLAN deployments. Artery's *artery.VanetNic* module is an INET 802.11 NIC configured according to [121, Table 9-138]. Also, IEEE 802.11's *TXOP limit* is set to zero for each AC explicitly. This enforces the full backoff procedure for every single packet.

Whether a 802.11 OFDM signal can be successfully decoded, is determined by the established NIST error model [58, 31]. This model gives success rates depending on the signal's MCS and the SNIR.

Both, signal and interference, are concurrent transmissions. Concurrent transmissions are distinguished only at the receiver's radio: *Signal* is the one transmission the NIC tries to decode while all other concurrent transmissions are interfering with this signal. The noise is modelled as an isotropic background noise of −104 dBm, which corresponds to the thermal noise at an ambient temperature of 21 °C and 10 MHz bandwidth, the typical ITS-G5 channel bandwidth (see equation B.3 in [29]). While a thermal noise floor of −104 dBm can be physically justified and is indeed also used by others [110], several authors of papers on IEEE 802.11p communication prefer a noise floor of −99 dBm[83, 100, 94, 116]. This higher noise floor incorporates further unwanted radiation and hardware imperfection.

As Bloessl and O'Driscoll [162] point out, choosing unreasonable parameters concerning the physical layer of VANET communication is widely spread. Unfortunately, also the prior default settings by Artery have been problematic with low background noise (−110 dBm) and a rather low receiver sensitivity (−89 dBm), *i.e.* signals below this sensitivity threshold can never be decoded successfully. Using this parameter combination, frames with a superior SNIR of 21 dB may get dropped in extreme cases. Bloessl and O'Driscoll propose to set both parameters to −98 dBm, which avoids rejected reception decisions by the PHY model despite a very good SNIR. Unrealistically large transmission ranges, which can be observed when the sensitivity alone is adjusted, are prevented by the NIST error model as an SNIR above 6 dB is necessary to achieve nominal error rates for frames sent at 6 Mbit/s.

In Artery, if the INET radio model is used, the proposed settings are adopted as follows for this thesis. Background noise is set to −98 dBm to represent thermal and receiver electronics' noise. The sensitivity of receivers is left at −89 dBm, though, based on observations from Fig. 3.11. The therein plotted graphs show the percentage of successful frame receptions depending on the

transmitter-receiver distance. Those frames are 500 B long and transmitted with 23 dBm power using QPSK 1/2 MCS (6 Mbit/s). A thousand frames are sampled for each distance step of one meter between receiver and transmitter. No interference is on the radio medium as only a single transmitter exists, *i.e.* the noise plus interference part of SNIR is solely the configured background noise.

As criticised by Bloessl and O'Driscoll [162], the NIST error model does not always provide a smooth transition between 'error-free' and 'total-loss' receptions. The frame delivery ratio does not drop all at once at a certain distance if the configured background noise and receiver sensitivity are configured close enough. This is the case for the two-ray interference model combined with −98 dBm receiver sensitivity in Fig. 3.11. Using −89 dBm sensitivity instead, it is not the SNIR error model but only the receiver sensitivity affecting the reception decision. Hence, one can see the sharp decline at about 970 m distance in Fig. 3.11. However, the small-scale variation feature by GEMV² is also an appropriate tool to smoothen these transitions. LOS links in GEMV², which are attenuated by the two-ray interference model in the first place, are superimposed with 1.5 dB of normally distributed power attenuations. Own field experiments and publications by others [65] confirm that IVC communication is indeed possible well up to 1 km under favourable LOS conditions using 23 dBm transmission power. The LOS model of GEMV² is configured to drop any frames beyond a distance of 1500 m, so excessive transmission ranges cannot creep into the experiments. The configuration of these GEMV² range limits is further discussed in Section 5.1. Receivers with a sensitivity of −98 dBm will be cut off abruptly at this distance, while with a sensitivity of −89 dBm the frame delivery ratio has already faded out. Hence, a receiver sensitivity of −89 dBm is advised for the employed radio model and used as default setting throughout this thesis.

Though concurrent transmissions 'collide' at the receiver's antenna, *e.g.* because of the hidden station phenomenon known with CSMA/CA, a packet may still get received successfully if its strength is well above the summed strength of interfering transmissions. Furthermore, some 802.11 chipsets are known to 'capture' signals, *i.e.* they can switch to a stronger signal even if another reception is already ongoing [34]. By default, INET's 802.11 receiver continues decoding its present signal. Strong concurrent transmissions (starting slightly later) are thus strong interferers and decoding likely fails. Artery ships with a customised *artery.VanetReceiver* model reflecting the described capture effect. This behaviour is not demanded by implementations in the 802.11 specifications, but often observed with real chips [42]. The model's behaviour can be configured via the capture threshold parameter, which affects the receiver's reluctance to switch to another signal: Only if the signal's SNIR exceeds the threshold, it will stop decoding the current signal and switch to

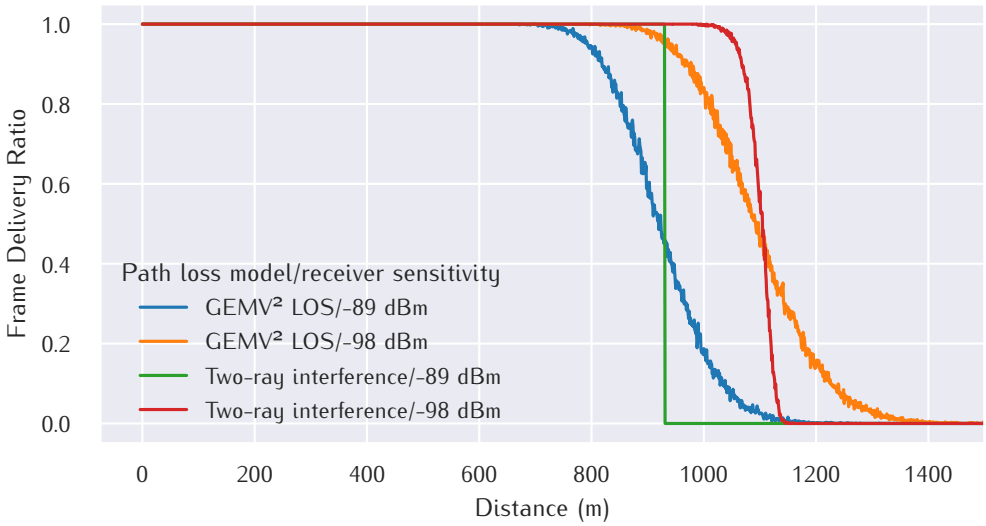


Figure 3.11: Frame reception rates over distances using 23 dBm transmission power with Artery's radio model

the stronger one. Capturing a packet transmission, however, does not imply a successful reception because the SNIR-based error model ultimately decides on the packet reception probability.

Reporting Channel Busy Ratios

DCC heavily relies on CBRs measured by the local radio. Some (optional) sub-layers such as DCC on the network layer, even share these local measurements with neighbouring stations. While a conventional WLAN NIC does not report CBRs to upper layers, this feature is a crucial element in VANET deployments.

ETSI [128] defines CBR as the time portion during the last 100 ms in which the strength of received signals exceeded −85 dBm [128]. It is thus an indicator on the channel usage by surrounding stations. In contrast to the 'busy state' tracked by 802.11's MAC, CBR does not count for the duration of own transmissions and ignores the Network Allocation Vector (*NAV*) duration, a virtual carrier-sensing mechanism relying on the duration field carried by 802.11 frames. Thus, CBR is closely related to the physical carrier sensing, *i.e.* CCA based on energy detection. The given threshold of −85 dBm is also in line with the minimum receiver sensitivity required by IEEE 802.11 for OFDM transmissions on 10 MHz channels [121, section 17.3.10.6].

This demanded minimum sensitivity is usually excelled by current receivers, *e.g.* Cohda Wireless [141] claims a receiver sensitivity of −99 dBm. A similar value of −96 dBm can be found in an undisclosed datasheet of a competitor. While the original INET receiver model possesses parameters to configure the sensitivity levels, it dramatically affects the measured CBR values if the MAC's reception state is used for CBR calculation. The boxplots in the left column of Fig. 3.12 show this: Receivers with high sensitivity

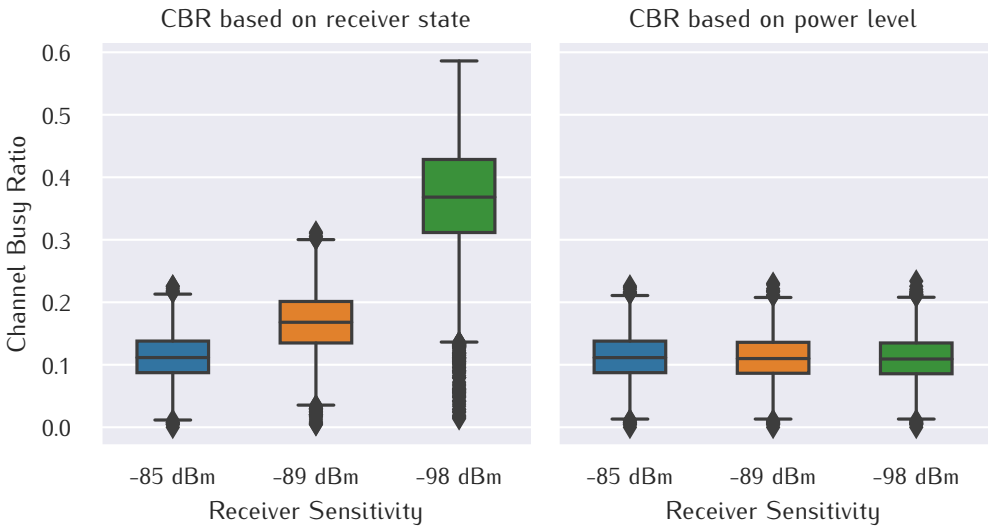


Figure 3.12: Boxplots of CBRs calculated by receiver state or received power level

are more often receiving frames and thus in a non-idle state. Hence, CBRs calculated on those non-idle periods are exuberantly large. The boxplots in the right column show Artery's tweaked receiver model which evaluates the power levels of signals. Clearly, the tweaked model reports CBRs consistently while still allowing more sensitive receivers to decode weak signals.

If more sensitive receivers treat the medium as busy in CCA based on their individual performance, they are discriminated by the backoff procedure compared to less sensitive receivers. Latter ones detect the medium less often as busy and thus reduce their backoff counters more frequently. Sensitive receivers are then discriminated because they have to wait longer periods until gaining channel access. This unfairness among stations with differing sensitivity levels can be avoided by a fixed CCA threshold independent of the receiver sensitivity. While receiving a frame with a signal strength below CCA threshold, the station may still decide to start a transmission on its own. Artery's aforementioned receiver model realizes exactly this fair behaviour. Please note, that group-addressed QoS data frames in a VANET (the vast majority of frames) have their duration field set to zero [121, section 9.2.5.2] and thus virtual carrier-sensing (NAV) is disabled for those transmissions.

3.2.4 ITS-G5 Network Protocols

ITS-G5 protocols comprise some non-trivial entities such as GN, DCC, and the security entity. To the best of my knowledge, my open-source implementation of the ITS-G5 protocols named 'Vanetza' remains the only free and feature-complete realisation as of today. While not every specified feature of ITS-G5 is implemented, Vanetza can still be coined feature-complete

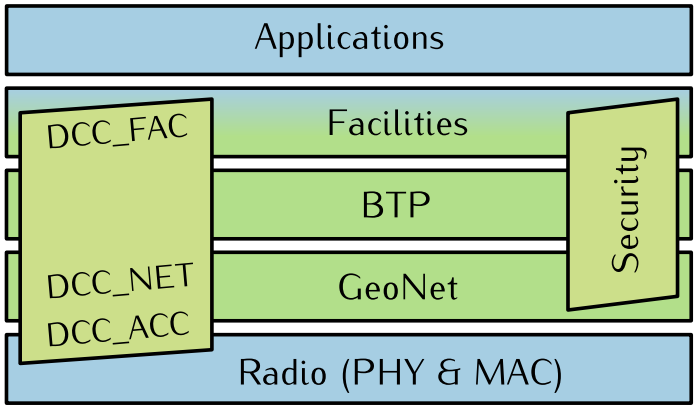


Figure 3.13: Layers supported by Vanetza (green) in the ITS-G5 architecture. First published in [12].

because it comprises all required features to operate a C-ITS station as described in C2C-CC's Basic System Profile [139] and the related Delegated Act [177].

An alternative to Vanetza is the GeoNetworking stack written in Java by Alex Voronov [122]. However, this stack still lacks packet forwarding and congestion control mechanisms. Previously popular IVC software tools such as NEC Laboratories' C2X SDK [54] have been discontinued and are no longer available. The C2X SDK has been given away without royalty for research purposes and thus targeted a similar research-affine audience like Vanetza. OpenC2X is an active project offering an open-source software environment for V2X experiments and prototyping in the context of ITS-G5 [118]. However, it is not specifically designed for being used in network simulations but on (embedded) hardware. Also, OpenC2X's network capabilities are lacking because it does not implement any packet forwarding.

A design criterion for Vanetza has always been [1], that it shall not be a mere abstracted model of ITS-G5 but can also run as a full-grown stack on embedded hardware. Hence, all packet structures by Vanetza comply strictly to specifications leading to binary-compatible over-the-air packets. As a side benefit, quantisation errors of header fields are correctly modelled in simulation runs then as well. For example, position data in GN headers can only represent tenths of micro degrees and packet lifetimes can only be encoded in steps of 50 ms at best. As highlighted in green colour in Fig. 3.13, Vanetza implements the ITS-G5 specific layers only. In particular, it does not implement the physical and link layer specified by IEEE 802.11 [121]. Support for the Facilities layer is rudimentary, *i.e.* Vanetza ships with support to encode and decode ITS messages such as CAM and DENM by integrating *asn1c* [210]. However, Vanetza does not fill those messages on its own as this requires access to platform-specific data such as vehicle signals.

Special care is necessary with respect to timing components, e.g. GN beacon timers, CBF buffering, expiry of LocTEs etc. Instead of using one particular clock and thus imposing a particular runtime context or operating system, components of Vanetza rely on `vanetza::Clock` and the associated `vanetza::Runtime` interface. Former is very similar to C++ clocks of the `std::chrono` namespace with an epoch starting at 1st January 2004 midnight International Atomic Time (*TAI*) and steadily increasing since then, i.e. following TAI without leap seconds. While `vanetza::Clock` defines types to represent durations and time points, it does not report current time via a static `now()` method. Instead, instances of the `vanetza::Runtime` interface are aware of the current time point and allow scheduling of arbitrary callbacks. As a typical example, a packet buffer schedules a callback to remove a packet from its storage after lifetime expiry. Implementations of `vanetza::Runtime` exist for manually stepping time during unit test execution, relying on simulation time from OMNeT++'s event scheduler or actual system clocks. This approach enables deployment of Vanetza in a wide range of runtime contexts and optionally decouples Vanetza's timing from wall-clock time to run it faster (unit tests) or slower (large simulation networks) than real time.

GN relies intensely on position calculations for packet forwarding, *e.g.* distance between network nodes is a direct input to determine timeout spans of CBF. Vanetza employs *GeographicLib* to avoid inaccuracy at these calculations. GeographicLib's documentation claims its accuracy to be 'close to round-off, about 5 nm' [132]. For checks, if a particular position lies within a given destination area, coordinates are converted via GeographicLib to a local Cartesian coordinate system first. The origin of this local tangent plane is the centre of the respective destination area. This method is also recommended by the BSP [139].

Vanetza also integrates OpenSSL and Crypto++ as two backend implementations of cryptographic operations. When security features are enabled, the network layer will encapsulate outgoing packets in a secured message envelope. This envelope contains a signature for this individual packet content based on the Elliptic Curve Digital Signature Algorithm (*ECDSA*). Receiving stations can then verify the integrity and authenticity of the packet by checking the signature and the accompanying certificate. The full certificate may be contained in the secured message, but due to its length it is omitted regularly. In such cases, only its digest is used to identify the full certificate. If all stations know certificates of each other, this digest is enough for secure communication. Since the set of communicating stations is constantly changing in a VANET, every 1 s the full certificate is attached to a sent CAM unsolicitedly. Furthermore, stations can request the full certificate and even the certificate chain from other stations explicitly.

Concerning the ITS message length, the overhead induced by the certificates is significant. Thus, a simulation environment shall realistically model the length of packets as it causes longer air durations and ultimately contributes to higher CBRs. In contrast to packet lengths, the computational overhead is an unwanted effect when evaluating network performance. In a prior publication [7], it has been shown how the computational complexity of cryptographic operations can be avoided in simulations while still maintaining fully compatible network packets. Under the assumption that simulation's focus is not on security, a dummy signature can be inserted in the packet headers instead. Receiving stations will accept this dummy signature without proper cryptographic verification. This approach conserves the communication overhead in terms of message lengths but dramatically reduces the computational burden caused by cryptographic operations. The only prerequisite for this approach, the absence of any malicious station, can be guaranteed as every simulated station is under experimenter's control. However, experimentalists can still use the full security pipeline if they wish to do so.

3.2.5 ITS-G5 Facilities and Applications

Facilities and the applications built upon them play a decisive role in performance evaluations of ITS-G5 networks. Without these services, no data is transmitted, and no channel resources are consumed at all. Modelling ITS applications with their typical communication demand is thus a vital aspect.

It is one of Artery's strengths that stations with multiple concurrent services as well as different assemblies of services can be modelled. Besides a number of services shipped with Artery, prototypical services can be realised with less effort [16]. A good hint regarding the set of mature ITS services is the list of well-known Basic Transport Protocol (*BTP*) port numbers given in [176]. On the basis of this list, this section presents a sub-set of services which is believed to have a significant impact in early deployments.

Cooperative Awareness

The CA service is the primary source of packets in an ITS-G5 VANET because it continuously generates CAMs. Each vehicle equipped with a CA service generates one to ten CAMs per second depending on its own dynamics and the prevalent channel congestion. The CAM generation rules given by [171] are implemented by Artery's `CaService`. Vehicle data originating from SUMO are evaluated every 100 ms if any of the following three vehicle dynamics rules triggers:

1. heading has changed by more than 4°

2. current position differs by more than 4 m from the one encoded in the last CAM

3. absolute speed difference since the last CAM exceeds 0.5 m/s

Whenever one of these rules triggers, the determined CAM update rate is kept for the following three CAMs. Hence, the CA service does not immediately reduce its generating rate even if its vehicle dynamics are low then.

Furthermore, DCC has an explicit influence on the packet generation rate. While the triggering rules mentioned above describe the desired packet generation rate by the CA service, this rate may get limited by DCC imposing a lower rate. However, at least once per second, a CAM will be generated anyway. Thus, the effective message rate ranges from 1 Hz to 10 Hz.

The length of a CAM also varies over time as its low-frequency container is only appended every 500 ms. This low-frequency container includes data fields changing only slowly such as vehicle's role (*e.g.* if a police car is in action or only an ordinary road participant) and state of exterior lights. `CaService` reflects this 'breathing' of message length accordingly. Typically, the length of CAMs generated by `CaService` ranges from 41 B to 242 B.

Empiric CAM statistics published by C2C-CC [151] indicate an average length of about 350 B for a CAM packet, *i.e.* including security and other headers from lower layers. Their findings also show that individual manufacturers' preferences on including particular optional fields affect the message lengths considerably.

Decentralized Environmental Notifications

In contrast to CAMs, the occurrence of DENMs cannot be predicted because they are always linked to events. Typical use cases of the DEN service include adverse weather conditions, dangerous situations because of hard braking, stationary vehicles such as broken-down or crashed vehicles, and traffic jams.

Many triggering conditions are way beyond the scope of network simulations, though. Obviously, it is not feasible to simulate the weather additionally. However, we can take advantage of what all DENM have in common: They begin at a certain time point and are valid for a limited period and region. With Artery's *storyboard* feature one can define such external circumstances dynamically as a Python script, which is executed by the network simulation. Within a *storyboard* script, conditions are defined that ultimately trigger use cases of individual vehicles' services. These conditions can limit the time window, the applicable map region, and the set of vehicles for which an event can occur. Furthermore, dynamic parameters of vehicles can be taken into account, such as a vehicle's speed or the speed difference to surrounding

vehicles. Especially for safety-critical use cases, the evaluation of TTC can be handy. For details about the mechanisms of the *storyboard* you should refer to its introduction in [11] or the detailed example elaborated in [16].

The variety of use cases associated with one DEN message type is also reflected by the `DenService`. This service deals with the commonalities of those use cases, *i.e.* that all DENMs are disseminated as GBC packets and contain an *ActionID* field identifying the local station and the particular event by a sequence number. Peculiarities of each use case are handled by dedicated sub-modules of each `DenService`. Each use case sub-module evaluates its specific triggering conditions per vehicle update cycle and generates a draft of a DENM, which is completed by the hosting `DenService`. Such a DENM draft contains the cause code, destination area, and validity duration for the detected environmental situation. Furthermore, the packet priority and repetition pattern differs among use cases and thus need to be set by the respective sub-modules.

Cooperative GNSS augmentation

Cooperative GNSS augmentation aims at enhancing the positioning accuracy by providing information to eliminate common error sources in satellite positioning. As presented by Speth et al. [9], a parking vehicle may gain a very accurate positioning solution over time as its position is known not to change. Determined common errors, *i.e.* those experienced equally by all (also moving) GNSS receivers located in the same region, are then shared by stationary vehicles. While the study in [9] focussed on parking vehicles for deployment, RSUs can employ this service as well if they are also equipped with a GNSS receiver.

The spread information ages quite fast because the underlying correction algorithm use double differences. This implies the necessity to update the message rapidly and a limited packet lifetime. As outlined in [9], the lifetime should not exceed 5 s. The message format is a concatenation of several message types from Radio Technical Commission for Maritime Services (*RTCM*) 1040 [80]. These sentences comprise observation data for GPS and GLONASS, two particular GNSSs, as well as a description of the stationary receiver's antenna and a reference position. In the end, this RTCM ITS message has a length of 372 B at the application layer. This length is equivalent to $N_{gps} = 10$ GPS and $N_{glo} = 10$ GLONASS satellite observations, as printed in Table 3.3. Message parts with non-integer byte lengths are padded to the next byte boundary. The message is designed for dissemination up to a distance of 5 km away from the emitting station.

Typical DEN use cases, as specified by the C2C-CC, address destination areas with a radius of 1000 m at maximum. GNSS augmentation data, as outlined in Speth et al. [9], conveys information applicable for larger areas.

Table 3.3: Length of RTCM-based message

Message part	Fixed	per Satellite
Preamble	3 B	NA
Checksum	3 B	NA
Station coordinates	21 B	NA
Receiver antenna description	9 B (+ strings)	NA
GPS observations	8 B	$N_{gps} \cdot 15.625$ B
GLONASS observations	7.625 B	$N_{glo} \cdot 16.25$ B

Hence, this service stresses the network performance with respect to routing of long-range GBC packets. Attention is required, though, as such large destination areas exceed GN's maximum area size allowed by default. One must adjust the GN protocol constant `itsGnMaxGeoAreaSize` accordingly, whose default value is 10 km². A limit of 80 km² works fine for this GNSS augmentation service.

In recent standardisation of infrastructure services [144], a similar service called GNSS Positioning Correction (*GPC*) is defined. While the GPC service's message format is also derived from RTCM, the encoding has been transformed to Abstract Syntax Notation One (*ASN.1*). Also, its dissemination range defaults to only 400 m. In the following, we use our custom RTCM service as it adds long-range routing to the overall network usage.

Collective Perception

The motivation behind Collective Perception (*CP*) is the wish to overcome the limited field of view and range offered by local sensors. CP shares the locally perceived objects with other stations nearby. Sharing lists of objects consumes considerably less bandwidth than sharing raw sensor data. Beside object lists, a CPM also contains the reference position of the perceiving vehicle as well as a description of the sensor capabilities. Knowledge of sensor capabilities such as field of view and range then allows one to determine the absence of objects. [130, 5, 6, 175]

Initial studies considered various flavour of CP, *e.g.* extending the CAM format instead of employing a dedicated message type. Ongoing standardisation centers around a dedicated CPM. Referring to Günther et al. [6], a CPM is generated by the same triggers as a CAM. *TR 103 562* [175] allows generation rates between 1 and 10 Hz, independent from CAM rates. In particular, a full CPM comprising objects and sensor information is generated at least once per second. CPMs generated in between can employ a reduced set of containers to mitigate channel occupancy, *i.e.* these omit the rather static sensor information.

Artery has been enhanced by an environment model during the work on [6]. This environment model introduces the following features:

- Sensors with a given range and opening angle can be attached to vehicles at each side.
- Other vehicles within a sensor's field of view are added to the hosting vehicle's object list.
- Line-of-sight obstructions by other vehicles and buildings are considered.

By utilising this environment model, CPMs can be generated, which contains the actually visible objects. Hence, a CPM varies in size depending on its own sensor capabilities and the current surroundings.

In the context of this thesis, an implementation of CP called `CollectivePerceptionMockService` is employed. This service is simplified with respect to the encoding of the CPM. Instead of ASN.1, OMNeT++'s packet format is used, which is easier to handle in simulation. Yet, the presence of each container and object contributes to the length of the simulated CPM. Consequently, channel resources are utilised by this simplified service in a quite similar way.

If coupling between the generation of CAMs and CPMs is desired, the `CollectivePerceptionMockService` listens for the CAM transmissions by a sibling `CaService`. Upon CAM generation, a CPM will be generated based on the current view on the environment. Optionally, this generation can be delayed by a predefined period to spread channel access timing. If not specified otherwise, CPM generations are delayed by 50 ms. This leads to a uniform CAM + CPM packet generation pattern of 20 Hz if the CA service is operating at its highest generation rate of 10 Hz. Alternatively, `CollectivePerceptionMockService` can also generate its messages independently. In this case, it will emit a new CPM at a predefined rate, *e.g.* 10 Hz, as long as DCC's TRC does not throttle this rate.

Based on the numbers given in Günther [130, section 5.3], the message length is between 37 B to 709 B. The minimum length represents a CPM, where only the originating vehicle is described, *i.e.* its position, heading, speed and dimensions. For each sensor's field of view an additional container with a length of 9 B is added. Each included object accounts for further 19 B to 29 B. This size varies with the inclusion of optional object attributes such as its heading, length, width, acceleration and type. While an object's speed and distance are always included, availability of the mentioned attributes depends on the particular sensor's capabilities.

Infrastructure Services

Services operating on infrastructure, *e.g.* by an RSU attached to traffic lights, are not in the main topic of this thesis. However, due to the fixed-position of RSUs, their messages can cause local 'hot spots' in terms of channel

load. While the content of infrastructure messages is not modelled in the present version of Artery, their impact on local channel congestion can still be modelled.

Two prominent message types are MAPEM and SPATEM, which belong to the RLT and TLM service, respectively [144]. These services work in tandem: A MAPEM describes the road layout of an intersection with its conflict zones, ingress and egress lanes. A SPATEM builds upon this particular topology information and adds signal and phase information of the associated traffic lights. Both message types are disseminated as GBC packets, addressed to circular destination areas with a radius of 400 m. Likewise, both are generated at a rate of 1 Hz.

Following the instructions given in [177, Annex 3], a SPATEM for an intersection with four ingress lanes and only mandatory fields set results in a message with a length of 49 B. By adding speed advises for 'green wave', the length grows to 60 B. The definition of SPATEMs includes many fields and containers of variable length, so that the message length can become considerably larger. For example, a SPATEM containing a more detailed signal time plan with the next ten signal changes for each lane is already 252 B long.

This variability in message lengths depends not only on an intersection's topology but also the level of detail included by the road operator. Thus, it is almost impossible to determine a universally valid message length in this context. Since this thesis is mostly interested in the channel load caused by SPATEMs and MAPEM, it is sufficient to let RSUs generate mock-up messages. The hierarchical structure of SPATEM and MAPEM, *i.e.* optional containers contain further lists and containers, suggests an approximation of their message lengths by log-normal distributions.

Mimicking the channel load induced by traffic infrastructure works as follows. First, an RSU is placed at a regulated intersection. Each RSU is then equipped with two `InfrastructureMockService` instances. `InfrastructureMockService` generates GBC packets addressed to the circular destination area centred around the hosting RSU at a fixed rate. Radius, generation rate, packet priority, message length are configuration parameters. By default, a radius of 400 m is used, and a packet is generated once per second. The packet priority is set to DCC Profile (*DP*) 1 (AC_VI) for the TLM mock and DP 2 (AC_BE) for the RLT mock services, as specified in [144]. During service initialisation, the (real) message length M is drawn from the specified random distribution and remains fixed for the particular RSU during simulation. A log-normal distribution with parameters $\mu = 5.4$ and $\sigma^2 = 0.09$ is used by default. μ and σ^2 are the mean and variance of the logarithm of the message length $\ln(M)$. The average length resulting from this distribution is 257.24 B. Since only an integer number of bytes m can be encoded, the drawn length

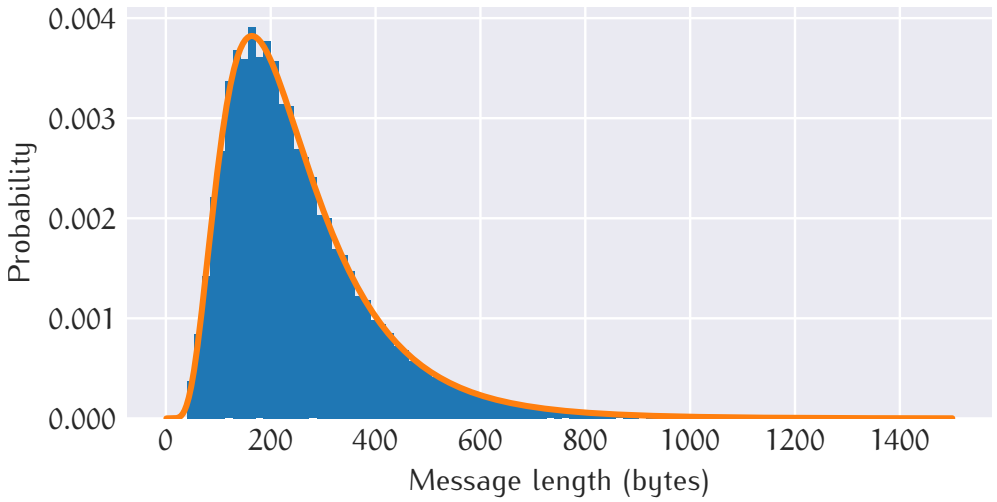


Figure 3.14: Log-normal distributed message lengths for SPATEM and MAPEM

is truncated as $m = \lfloor M \rfloor$ and clamped to the interval $[40, 1500]$. Figure 3.14 plots the probability density function (orange) of the described log-normal distribution, as well as the histogram of 10^6 randomly generated message lengths (blue).

3.3 SELECTION OF METRICS

Metrics are a prerequisite to compare the achievable QoS in ITS-G5 setups. Without those metrics, it would be impossible to quantify the impact on communication performance of the studied variants. This section gives an overview of found IVC performance metrics and discusses their implications.

Ideally, the employed metrics cover how good typical requirements of V2X applications can be fulfilled. Referring to the applications presented in Section 2.1.1 and Section 3.2.5, requirements can be clustered:

1. **Information Age** Periodic services prefer a short retention time, so that the information age on the receiver side is low. Higher message rates improve messages' usability, *e.g.* more precise predictions regarding the state of nearby vehicles. Especially vehicles in immediate vicinity benefit from high rates and short latency, *i.e.* the importance of these aspects decrease with rising vehicle separation.

2. **Emergency** Emergency use cases expect almost instant channel access to prevent any imminent harm. However, these messages should not occur regularly but only in critical traffic situations. Reliability is also important to the extent that all affected vehicles have the conveyed information available. The range is usually only a 'shortly before crash' distance of a few tens of metres.

3. **Areal Dissemination** Event-based services such as warnings about adverse weather conditions or traffic jams ahead, may affect vehicle safety, but the reaction time is more relaxed in comparison to emergency events. With growing distance to the event location, also the available time window to react increases. Dissemination of event information shall not be delayed in the order of seconds, but a few hundred milliseconds still give vehicles plenty of time to react.

4. **Temporal Availability** As an extension to the initial dissemination phase of event-based services, some services put emphasis on reaching as many vehicles as possible over long distances and comparatively long periods. Their impact on the channel congestion should be kept low, *i.e.* delayed delivery is acceptable to avoid frequent retransmissions.

In a nutshell, applications need to reach their respective audience in time. The time frame is depending on the distance between the sender and its audience and the allowable reaction time. Larger time frames and validity periods of conveyed information shift the emphasis from low-latency, short-range communication to delay-tolerant, mid-range communication. This shift is often reflected by the switch from SHB to GBC dissemination. In the following, metrics quantifying the performance of the dissemination process are thus discussed. As the the available resources for message dissemination are constrained by DCC, it is reasonable to assess the performance of congestion control explicitly with dedicated metrics.

3.3.1 Message Dissemination Metrics

End-to-end latency, *i.e.* latency measured from the time point an application generates a message until it is information is available for the receiving application, is essential for safety applications. An excellent example is the exchange of impact reduction containers, which are triggered when a near-crash situation is detected. However, latency is mainly critical in the near field. Measuring the mean latency is thus not beneficial, as it neglects the application requirements in a specific context. For example, asserting the fulfilment of latency requirements within a range limit is reasonable.

PER is defined as the ratio of faulty packets to all received packets. By definition, this metric does not apply to a whole network but needs to be computed by each receiver individually. This metric is common to assess receiver performance, *e.g.* by determining the maximum distance between a transmitter and receiver where the PER remains below 10 %. However, in large networks with many distant transmitters, the explanatory power

of PER is limited. Especially with GN forwarding schemes, corrupt packet receptions from remote transmitters may increase PER, but the information is still conveyed by closer forwarders.

Link-layer metrics such as PER or the number of transmissions are not reflecting the availability of information adequately. Similarly, PDR – the ratio of actually received packets to all transmitted packets – describes the reliability of communication links but hardly the reliability of information dissemination with forwarding. Duplicate receptions are not adverse by themselves either, and thus counting them is unappealing. Their negative impact on congestion is covered by DCC metrics, as described later on.

Two dimensions of information availability have to be considered: Spreading information throughout the addressed area and spreading information over time. Former can be measured by calculating the coverage of a GBC packet, *i.e.* how many of all potential receivers within the destination area have actually received the message at least once in a given time frame. Considering the standard hop limit and maximum duration of CBF timers, a time frame of 1 s is suitable. This time frame limits the coverage metric then to a single CBF dissemination phase, *i.e.* without subsequent repetitions or SCF. A network's capability to keep information available over time can also be determined by calculating the coverage, but using a significantly larger time frame, which may go up to the packet's lifetime. Though vehicles may leave the destination area, the denominator of the coverage ratio can only grow over time as more unique vehicles visit the destination area. For very large destination areas, it can also be beneficial to subdivide coverage into rings [9], especially if the original sender is located at the destination area's centre. The ringed coverage captures then the information availability dependent on the remoteness of receivers.

Since CA is a major reason for messages in ITS-G5 networks, it is worth to look at this service specifically and periodic services in general. The generation rate of CAMs is affected by vehicle dynamics as well as throttled by DCC. As a consequence, the Inter-Reception Time (*IRT*) between consecutive messages is not necessarily a result of congestion control. Low vehicle dynamics, *e.g.* vehicles waiting at a stop line, are an equally valid reason for longer time gaps. Calculating the IRT for CAMs is not uncommon, though it is called differently sometimes. Autolitano et al. [81] refer to it as 'update delay', for example. An application-specific metric for CAMs and its American counterpart BSM is the calculation of the tracking error [95]. For this metric, each station extrapolates the positions with constant speed and heading known from the last received CAM (or BSM). The tracking error is the difference between this extrapolated position and the actual position. In field tests, such a metric is hard to realise because precise positioning of all stations is required all the time. In simulations, the global knowledge

of all vehicle positions can be gained easily. However, position updates are reported by SUMO only at discrete, fixed time intervals. Reducing these intervals more than absolutely necessary for CA triggering conditions, *i.e.* less than 100 ms, increases the simulation's computational load significantly. Especially in large simulation scenarios, precise and continuous position data is a challenge.

Boban and d'Orey [113] propose the 'Neighbour Awareness Ratio' as CA metric. This metric is similar to the aforementioned GBC coverage as it correlates the number of CAM receivers within a predefined range to all potential receivers. As CAM receiver counts any station that has received at least one CAM in a given time window, Boban and d'Orey use 1 s. However, this metric alone neglects the time between CAM updates and thus is less expressive than the tracking error. In combination with IRT, the time between two CAM receptions from the same source, the expressiveness can be improved. Plots by Bansal et al. [95], who calculated tracking error and IRT, also suggest a correlation between both. Application-specific metrics such as the tracking error are also susceptible to measure not only the network's QoS but also the quality of application decisions, *e.g.* when is a message generated and what is included. The pair of coverage ratio and IRT, however, can capture the QoS of arbitrary periodic services.

Gómez and Mecklenbräuker [116] deal with VANET metrics focussed on QoS of safety applications. In essence, they combine the cumulative density function of the end-to-end delay with the achieved dissemination range. They label emergency communication as 'reliable' if 90 % of the stations have received the emergency message before its 100 ms deadline. Comparisons between DCC variants are carried out by referring to both, the percentage of nodes within the given deadline and the maximum distance of these from the sender. Notably, Gómez and Mecklenbräuker restrict safety communication to single-hop packets.

3.3.2 Congestion Control Metrics

With channel congestion in mind, the measured CBR by each receiver is the prime metric to assess channel usage. As outlined earlier in Section 2.4, the channel usage shall remain below ~70 % as the risk for packet collisions increases tremendously beyond this threshold. Lower channel usage is an indicator of less efficient usage of resources by wasting capacities. Of course, low channel load is fine if no transmissions have been throttled as it may occur in scenarios with low vehicle density.

Aygun, Boban and Wyglinski [112] aim to capture the amount of unwanted interference by a metric that relates the number of receivers which are beyond a target distance to the number of all receivers of the same message. Though

the authors have the transmission power adjustments for CA originally in mind, it is conceivable to translate this metric to GBC dissemination. Especially at the fringe of a destination area, many stations can be affected by packet forwarding though they are outside of the addressed area. For these stations, the packet forwarding has no value and is thus as bad as interference. However, the outlined metric accounts only decodable packets and thus corrupted yet interfering receptions are neglected. If a novel forwarding approach considering interference by forwarders at the fringe is to be evaluated, the noise and interference power level should be employed as a measure instead. Simply reducing the transmission power at the fringe to keep interference low for non-participating stations is too short-sighted: With omnidirectional antenna patterns this can also adversely affect the forwarding success within the destination area. Interference exceeding the threshold of −85 dBm is ultimately included by the CBR metric as outlined in Section 3.2.3.

Autolitano et al. [81] add probes to DCC state machines to study congestion control. In particular, they track how many vehicles are in a particular state, the average permanence in each state, and the average number of state switching events per minute. These metrics give insight into the mechanics of reactive DCC. However, they are not applicable to adaptive DCC, which does not employ a state machine. Similarly, observing stations' allowed duty cycle as determined by adaptive DCC is not generally available among DCC algorithms. The compulsory pause T_{off} enforced by DCC_ACC after each transmission is available for any TRC variant.

ETSI has published a technical report where three key performance indicators for DCC algorithms are defined [96]:

- Fairness, *i.e.* channel access time of one-hop neighbours does not vary by more than 10 %
- Channel Load, *i.e.* CBR target is never exceeded by more than 10 %
- Stability, *i.e.* the gradient of transmission data rate or power is not inverted by more than 10 % over a period of 10 CBR reports

Though fairness is a noble goal, roughly equal channel times fall short of assuring fairness if stations are not equal. For example, stations equipped with more ITS applications may still be treated fairly even if they have to defer some additional messages. The fairness also ignores the station's context, which may justify preferential channel access because of a safety-critical situation. Independent of a particular station's communication demand, fair treatment of stations should become manifest by similar T_{off} pauses, at least for non-emergency packets. The issues with the stability indicator is that it has to be evaluated in a stable environment. While vehicles are moving, connectivity among them is in flux, and radio propagation conditions change

permanently. Thus, these key performance indicators are suitable to check a DCC algorithm, whether it has the demanded properties at all in a constrained environment. In more realistic and dynamic environments, however, they are hard to evaluate reasonably except for the maximum channel load.

Two metrics mentioned by [96] are derived from LocT and can amend plain CBRs by describing the network context. One such metric is 'communication range', which is the receiving station's maximum distance to any other sender heard within the last second. The other metric is called 'neighbour density', which counts the matching LocTEs also considered as the 'communication range' maximum.

3.3.3 Conclusion

In summary, the most important metric of congestion control is also its main input variable, the CBR. Further metrics such as neighbour density and communication range attribute CBRs as they describe under which circumstances the respective CBR has been determined. Further DCC performance indicators such as fairness are tricky to capture. A hint towards fairly distributed channel access is the equal distribution of T_{off} pauses after transmissions. For adaptive DCC variants, the distribution of the duty cycle limit δ is an alternative indicator. Unfairness can be revealed implicitly by diverging dissemination performance of adjacent stations.

Dissemination performance of periodic services is a combination of their range and update rate. A measure can be either the coverage ratio, *i.e.* percentage of reached stations in a given radius, or the distance to the most remote station reached before a given deadline.

As of today, emergency messages in ITS-G5 are DENMs with an emergency cause code. Low-latency dissemination to a nearby audience is key in case of an emergency. Latency towards more distant vehicles or of non-emergency GBC packets is less of an issue. Their dissemination performance is better described by the coverage ratio, possibly in conjunction with a deadline.

The essential metrics employed later in Chapter 5 are thus CBR, end-to-end-latency, IRT and coverage. These metrics are intuitively understandable and can be applied to any ITS-G5 VANET. If the network's performance is evaluated in its entirety, the comparison of transmission and reception counts of particular message types can substitute the calculation of per-vehicle IRTs: Recognising more receptions at a stable amount of transmissions is equivalent to shorter IRTs on average. Furthermore, the network performance can be investigated from various angles if they are combined with deadlines or range limits.

3.4 SUMMARY

This chapter discussed the triad of metrics, models and scenarios to evaluate VANETs methodically, focussing on their QoS. Metrics are essential to quantify the effect of QoS algorithms such as DCC and their parameters. As pointed out in this chapter, specific metrics exist for some applications, such as the tracking error for CA, but these are naturally limited to their use case. This thesis approaches QoS for ITS-G5 with many use cases in mind and thus picks more generic metrics. In particular, latencies, message rates, and coverage ratios can be applied to a broad range of use cases. Requirements by specific use cases can be given as thresholds of these metrics, *e.g.* a minimum message reception rate or the maximum acceptable latency. The effectiveness of congestion control is best captured by CBR measurements, which directly reflect the channel load as perceived by a station.

Section 3.2 detailed the simulation model employed in this thesis in a bottom-up approach starting with the physical layer. In any heterogeneous environment, *i.e.* where vegetation, settlement or traffic density varies, the resulting signal attenuation of most radio propagation models is a compromise as they can only be tuned to a single environment. GEMV² surpasses them as it dynamically evaluates which radio channel characteristic dominates for each radio link. From the perspective of ITS use cases, it is reasonable to assume omnidirectional antenna patterns due to their versatility. Besides modelling those physical properties, a lot of effort has been put into modelling all the ETSI ITS-G5 layers: From radio devices to the network protocols up to the ITS applications. The integration of Vanetza into the simulation model of the network stack ensures an accurate behaviour of the standardised layers not found in any other known simulation model.

Urban, rural and motorway scenarios feature characteristic speed and traffic patterns, which feedback to the triggering conditions of ITS use cases. The LuST scenario based on the city of Luxembourg is a renowned starting point for simulations set in an urban environment. With DekiNet2, this chapter closes the gap for a rural scenario featuring a mix of towns, country roads and a crossing motorway. Various forms of vegetation, including farmland and forests, are covered by this 400 km² large scenario. Signal attenuation calculations benefit from the outlined post-processing steps for scenarios based on OSM data, such as LuST or DekiNet2: For one, touching polygons of buildings can be merged. Overlapping polygons representing vegetations are also normalised, so only one type of vegetation exists at each location.

With metrics, scenarios and models in place, state-of-the-art VANETs can now be simulated. These setups act as the baseline for comparison with the enhancements presented next.

4 DCC AND GEONETWORKING ENHANCEMENTS

QoS (see Section 2.3) in terms of hard deadlines and guaranteed bandwidth is impossible to realise with the given 802.11 radio technology in truly ad-hoc topologies such as VANETs. Considering the susceptibility to errors of wireless communications in general, it can be acceptable for many use cases that no guaranteed QoS exists. VANETs based on ITS-G5 can provide decent performance if their level of congestion is controlled. As long as network participants coordinate their channel usage, QoS properties such as low latency are very likely albeit not guaranteed. Research and standardisation efforts to facilitate such a distributed collaboration among network participants have been presented in Section 2.4.

Spending time with the development of Vanetza, the implementation of the ITS-G5 protocol stack, results in an exceptional involvement with the underlying specifications and their advancement over time. Experience from using Vanetza in various simulation experiments but also in field tests led to a bundle of ideas to further enhance these protocols. This chapter suggests improvements for the GN layer in Section 4.1 as well as the DCC cross-layer in Section 4.2. As these entities are found in every ITS-G5 station, any application, running on top of them, benefits from their improvement. The radio technology in terms of IEEE's 802.11 PHY and MAC remains unmodified, *i.e.* all proposed enhancement could be deployed by pure software updates and requires no hardware modifications.

The proposed enhancements are accompanied by simulation experiments, which demonstrate the difference between the standardised approach and the enhancement. Because the principal purpose of these simulations is the illustration of the original problem and the enhancement's qualitative impact, the employed setups are only briefly outlined. However, all of them follow the methodology presented in Chapter 3. A thorough evaluation of the proposed enhancements follows in Chapter 5, including a detailed breakdown of simulation parameters.

4.1 GEONETWORKING ENHANCEMENTS

ETSI divides the specification of GN into a larger document covering the 'media-independent functionality' [193] and supplementary documents dealing with 'media-dependent functionality'. Latter documents exist for

ITS-G5 [195] and for LTE-V2X [196]. As a consequence of this division, the majority of GN algorithms assumes that a packet will be transmitted practically without any further delay after it has been passed down to the link layer. Congestion control, which limits the throughput and controls the interval between transmissions, is only considered in the media-dependent supplement for ITS-G5.

Giving thought to the characteristics of the underlying access technology in the first place is a good idea when dealing with congestion control. However, limitations of the link layer are only rudimentary considered so far. Up to now, only a non-binding, informative appendix exists [195]. As Section 4.2 will show, the collaboration between DCC and GN can be improved considerably.

In this section, restrictions such as TRC imposed by DCC at the link layer are set aside. When GN passes a packet down to the link layer, this packet will be transmitted as soon as plain EDCA can grant channel access. Even under these idealistic conditions, which come close to the assumption by media-independent GN algorithms, GN does not perform as good as it could. Deficiencies have been found regarding the forwarding algorithm CBF, management of LocTEs and SCF. These deficiencies are identified in the following, along with enhancements to fix or at least mitigate them. Accompanying simulations highlight the original problem and show that the proposed enhancement is effective principally. Large-scale simulations using heterogeneous scenarios are postponed until Chapter 5.

4.1.1 Bounded Redundancy while Packet Forwarding

CBF is the default option for area-forwarding in GN. Routing decisions by CBF are truly distributed among network nodes and expected to be robust to cope with fast topology changes and radio propagation challenges found in VANETs. The fundamental idea of CBF is that a station refrains from forwarding when it has heard another station already forwarding the very same packet meanwhile. These mechanisms have already been introduced in Section 2.2.1.

Besides plain CBF the standard also defines an 'advanced' CBF variant with more features. Among the amalgam of features is the introduction of packet counters. Those counters are maintained by the receiving station for each unique, received packet. Instead of immediately stepping back from forwarding as in plain CBF, the receiver only refrains from further forwarding if the counter has reached a predefined threshold. The idea behind this mechanism is to increase the robustness of forwarding by increasing packet redundancy. Still, the amount of redundancy is controlled by counting duplicated packet receptions. Plain CBF, which does not explicitly use counters, should behave as if the counter threshold is set to one.

Problem Description

Two problems exist within the standard regarding CBF, though: First, no hint is given what a suitable value is for the counter threshold. Selecting a threshold is ultimately weighing up robustness versus channel congestion. Opinions on the required level of robustness are likely to differ considerably and may even need to get adapted dynamically. Second, the lifetime of packet counters is bound to the respective packets in the CBF buffer. Though the specification only vaguely states that 'a router in CBF mode maintains a counter for the number of retransmissions for a packet' [193], the accompanying diagram and pseudocode clearly indicate aforementioned relation between packet counter and CBF buffer. A station thus loses the packet counter as soon as it has decided to dequeue the respective packet, *e.g.* after forwarding the packet. Upon any follow-up reception of the same packet, the station has no notice if it has processed this packet before. Therefore the station will start a redundant contention phase unknowingly from the beginning. Failing to control redundancy is more critical than agreement on a particular redundancy threshold because this value will not be honoured anyway.

Figure 4.1 visualises the resulting flooding problem schematically, assuming a redundancy threshold of one. The green bars indicate the contention period of a buffered CBF packet by the respective stations A, B, or C. These stations have different distances to the original sender and thus their contention periods differ. Station C's CBF timer expires first and therefore forwards the packet. Only B hears this forwarding as station A is out of C's transmission range. B stops contention, discards forwarding and also the associated counter. The time difference between the initially scheduled CBF timeout and the premature removal of the packet by B is highlighted in yellow. Station B will again start forwarding contention upon reception of A's forwarding. Neither A nor C contend at this time anymore as they have already forwarded the packet. The loss of their packet counters then allows them erroneously to restart the forwarding process. Without a persistent packet counter, the stations cannot distinguish a first reception from on-going forwarding. This issue is exacerbated for plain CBF, whose overhearing relies only on the presence of a packet in its CBF buffer.

Problem Solving

The key to avoiding the described issue is to separate the packet overhearing from packet buffering. This separation can be achieved by decoupling the lifetime of CBF packet counters from the sojourn time of corresponding packets in the CBF buffer. For each distinct packet – identified by its source GN address and sequence number – these counters memorise the number of receptions. Those counters do not persist infinitely but can be removed

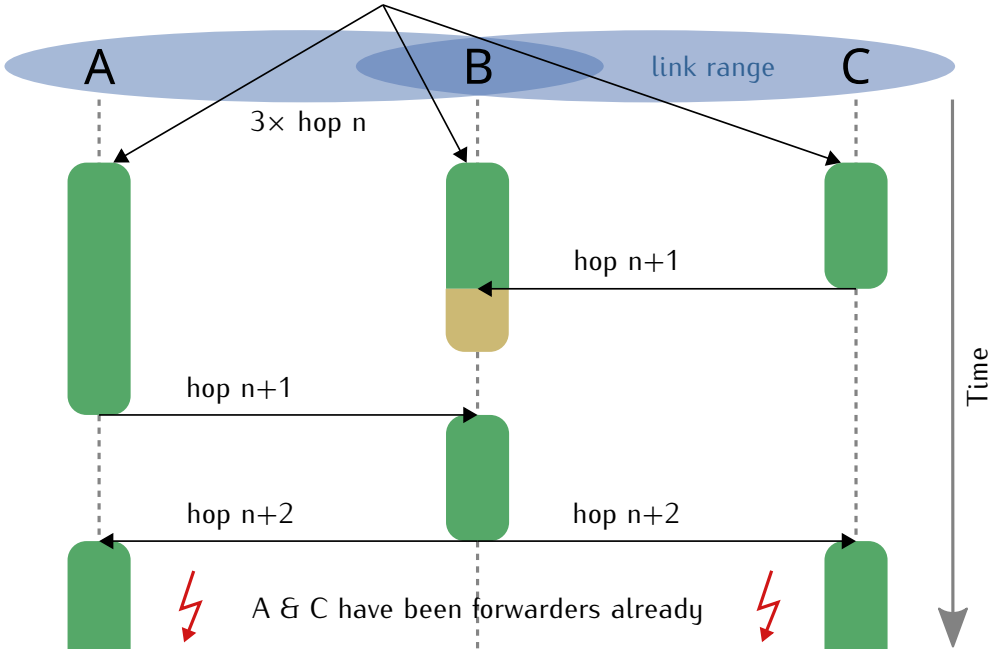


Figure 4.1: Flooding problem of CBF

after some time, *e.g.* after $itsGnDefaultHopLimit * itsGnCbfMaxTime = 1$ s. Consequently, a counter exists at least for the whole duration of one CBF dissemination phase. If such a dissemination phase has been interrupted, *e.g.* by SCF buffering, stations may participate again in dissemination when forwarding is resumed. As the described counters vanish after a predefined expiry, they are called 'fading' CBF counters.

While the 'advanced' CBF algorithm of GN already includes the examination of packet counters explicitly, the plain CBF variant needs a minor modification. The standard checks only if the CBF buffer `B` already contains the received packet `P` to control the overhearing. In the terminology of ETSI [193], the condition `P in B` needs to be extended towards `P in B OR counter(P) >= 1`. Support for redundant forwarding can easily be added to plain CBF by replacing 1 by the desired redundancy threshold.

Bookkeeping of already received multi-hop packets is not new in its own right. Paulin and Rührup [110] have also noticed that CBF is prone to unbounded repetitions and recommend to employ Duplicate Packet Detection (*DPD*) as a countermeasure. However, Paulin and Rührup do not further outline how to maintain the data structure for DPD. If the lifetime of counters is not limited then only a single, non-interrupted CBF phase is possible. Thus, quasi-persistent CBF counters may conflict with SCF, which resumes dissemination after some time.

Kühlmorgen et al. [200] propose to check for duplicate CBF packets at DCC_ACC. They intend to restore the overhearing mechanism of CBF, which fails when a packet is re-broadcasted while it is also enqueued by DCC_-

ACC for transmission. Since the duplication check by DCC_ACC takes place when GN no more buffers the CBF packet, the CBF buffer cannot be their reference. At least for plain CBF, their approach implicitly solves the flooding issue though they propose duplicate checks for a different reason. However, maintenance of DPL is not addressed by Kühlmorgen et al. either.

GN describes DPLs which are attachable to LocTEs, *i.e.* a dedicated list is tracked for each station. The length of each list is limited to $itsGnDPLLength = 8$ entries. Unfortunately, GN explicitly disables DPD if CBF is employed with the justification that CBF needs to handle duplicates on its own. As an alternative to fading counters, however, it is viable to realise the separation of counters and packet buffer via these DPLs. Each entry in a DPL is then also associated with a packet counter. The major difference to fading counters is how these counters are reset. If the capacity of *itsGnDPLLength* is reached, GN demands that the new entry replaces the oldest entry in the list. In contrast, fading counters do not disappear due to capacity limits, but only if their lifetime expires. While a packet's forwarding is ongoing, the respective counter's lifetime is extended on every counter increment.

With functional CBF counters in place, another deficit occurring with CBF can be fixed. A forwarder will discard a multi-hop packet immediately if the respective PDR limit is hit by the respective source station. If CBF is used for forwarding, the PDR tracked in the LocTE is erroneously updated upon each reception. Those receptions also include all received duplicates due to CBF by neighbours. The proposed remedy is to conditionally update the PDR only if the CBF counter is zero. Forwarding of excessively generated packets by any source station acting like a 'babbling idiot' will thus still be aborted, but protocol inherent forwarding operations are no more penalised.

Simulation

To shed light on the outlined flooding problem and to which extent the presented mechanisms can prevent it, a rather stressful CBF scenario is simulated. In the Griddy scenario, a constant set of 202 vehicles equipped with ITS-G5 is driving around in the investigated time window of 30 s. All of them transmit GN beacons, but only two are also transmitting GBC packets, which need to be disseminated by CBF to the vehicles within 1 km. The packet lifetime is set to 5 s so it does not limit the forwarding prematurely. If a vehicle is generating GBCs at all, it will do so in 3 s intervals. In order to emphasise the multi-hop dissemination performance, the transmission power is set to relatively low 10 dBm.

Table 4.1: Impact of CBF counters on dissemination of GBC packets

Experiment	Reached vehicles	Forwarding transmissions	Stopped forwardings		
			Hop limit	PDR	Outside
Standard CBF	1353	1186	4584	10194	338
Standard CBF without PDR	1833	2554	19582	0	800
Fading counters (threshold = 1)	1795	443	15	0	671
Fading counters (threshold = 3)	1829	1044	171	0	793
DPL counters (threshold = 1)	1795	443	15	0	671
DPL counters (threshold = 3)	1829	1044	171	0	792

This environment is studied for three realisations of CBF counters: Standard CBF, CBF with fading counters, and CBF with DPL counters. Table 4.1 also gives numbers for the standard variant with disengaged PDR limits. Variants using fading or DPL counters are presented for packet redundancy thresholds of 1 (classic CBF) and 3.

Standard CBF is basically flooding the networks in an uncontrolled manner. With PDR limits in charge, only 1353 vehicles are reached by all emitted GBCs, the lowest number of all experiments. Forwarding is mostly stopped because the PDR limit of 100 kB/s is hit, though the two source stations are generating packets only at about 170 B/s. Disabling the PDR limit shifts the reason to stop forwarding from PDR to the hop limit. Asides from CBF counters, the hop counter in every GN packet's header limits the ultimate number of hops. By default, a packet is discarded when it reaches the tenth station. Instead of the desired overhearing of CBF, it is this rudimentary hop limit that prevents even more network load.

Fading and DPL-based CBF counters lead to almost identical results. Both variants restore the CBF overhearing as the low numbers in the 'stopped forwardings' columns attest. The rightmost column counts those stations that refrain from forwarding, because they are located outside the destination area. They may still receive a GBC from a station located inside, though. This count is thus an indicator for GBC packets reaching the fringe of their destination areas. While the total number of forwarding transmissions is considerably lower with functional overhearing, it reaches these fringes similarly well as the dysfunctional standard CBF with its tremendous 2554 forwarding transmissions. An increased redundancy threshold shows only a minor effect on the number of reached vehicles, though the number of forwarding transmissions increases as expected. The benefit of redundant forwardings may only pay off in more challenging radio propagation environments.

As outlined and demonstrated by simulation in this section, CBF as specified by ETSI [193] uses channel resources excessively. Dissemination of GBC packets is accompanied by broadcast storms which are only stopped by hitting the PDR or hop limit. Fortunately, the proposed changes affect only the router behaviour but not the packet format. A revision of the GN specification can thus address the described issue without breaking backwards compatibility.

4.1.2 Maintenance of Location Table Entries

Each station maintains a LocT in its GN router, which is its primary knowledge base for routing decisions. The principles of message dissemination using GN have already been outlined in Section 2.2. For each station, from which a packet has been previously received, the LocT maintains a corresponding LocTE. If a station has been heard directly, *i.e.* without intermediate forwarders, then this station is marked as a direct neighbour by setting its LocTE's *isNeighbour* flag. The distinction between direct and relayed links is made by the packet type: GN beacons and SHB packets are never forwarded so their reception clearly indicates a direct neighbour.

GN evaluates at two occasions if a LocTE represents a neighbour, which is directly reachable via one-hop communication. One is the aforementioned GF, GN's default option for non-area forwarding. The other is SCF, which checks for the presence of any immediate neighbours.

Also other entities, *e.g.* the DCC sub-entity at facilities layer (*DCC_FAC*) as outlined by Khan and Härri [148], rely on the accurate number of neighbours to allocate channel resources equally among them. Lyamin, Bellalta and Vinel [201] depend on the number of neighbours to configure the state machine parameters of their reactive DCC approach. Thus, algorithms also exist beyond standardisation for which the maintenance of LocTEs is crucial.

Problem Description

The GN specification checks the *isNeighbour* flag at various occasions as outlined before. On reception of a GN beacon or SHB packet, both are never forwarded by third-party stations, the *isNeighbour* flag is explicitly set for the respective source station. However, the specification lists no condition resetting this flag. A LocTE once marked as a neighbour will thus retain a set *isNeighbour* flag for its entire lifetime.

Lifetime is limited to $\mathtt{itsGnLifetimeLocTE} = 20\,\mathrm{s}$ counting from the creation of a LocTE [193]. Whenever the stored position vector is updated, the lifetime gets extended by another period of itsGnLifetimeLocTE. Consequently, a LocTE remains available as long as packets from the corresponding station are received. LocTE updates also occur for the source station of a

GBC packet, *i.e.* for stations which may be outside of the immediate radio range. The effective lifetime can thus become considerably longer than a single itsGnLifetimeLocTE period.

The severity of false *isNeighbour* flags depends on the particular algorithm evaluating them. Congestion control may choose subpar parameters if it misjudges the actual number of nearby network participants. SCF may decide not to buffer a packet though no station is in the immediate vicinity anymore. Most critical is probably the risk that GF selects a station as next hop, which cannot be reached contrary to the assumption based on the LocT.

Algorithm 1 is an excerpt of the specified GF algorithm [193, Annex E.2], slightly simplified by omitting SCF cases. GF scans the LocT for stations marked as direct neighbours. Among them, the most promising next hop is the neighbour located closest to the packet's destination from the local station's perspective. This criterion is also known as Most Forward within Radius (*MFR*). If a suitable next hop has been found, the packet will be sent specifically to this station as MAC unicast transmission. If no other station is known to be closer than the own position, referred to as Ego Position Vector (*EPV*), GF resorts to a MAC broadcast. GN foresees no recovery mechanisms if a unicast transmission fails. Because of missing acknowledgements, the sending station's MAC layer will only retry the transmission a few times until it gives up.

Algorithm 1 ETSI Greedy Forwarding (excerpt)

```
1:  NextHop ← BroadcastAddress
2:  MinDist ← DISTANCE(Packet.PV, EVP)
3:  for all LocTE ∈ LocT : LocTE.isNeighbour do
4:      OtherDist ← DISTANCE(Packet.PV, LocTE.PV)
5:      if OtherDist < MinDist then
6:          MinDist ← OtherDist
7:          NextHop ← LocTE.LinkLayerAddress
8:      end if
9:  end for
10: return NextHop
```

Imagine a scenario where station 'Alice' gets marked as a one-hop neighbour by station 'Bob' because he receives an SHB packet from her. When Alice is departing from Bob beyond their radio communication range, the LocTE kept by Bob about Alice still contains a set *isNeighbour* flag. This LocTE remains existent as long Bob receives packets generated by Alice, even though other stations are required as forwarders between them. When Bob now invokes the GF algorithm, he may erroneously select Alice as next hop for packets destined close to Alice's current position. Obviously, Alice will never receive Bob's transmission because they are now separated too far, and the routing procedures fail at this point ultimately, *i.e.* the packet will never reach its actual destination area.

Problem Solving

Mitigation of the described problem is possible via the following modification: Similar to LocTEs, the *isNeighbour* has to become 'soft-state', *i.e.* this flag is reset after a predefined timeout. The flag remains active as long as *isNeighbour* is repeatedly set by SHB or beacon receptions. If a former neighbour leaves the one-hop communication range, the *isNeighbour* flag is cleared after the associated timer's expiry. This modification requires only to adapt the maintenance procedure of the LocT. Any other algorithms, *e.g.* SCF and GF, need no adjustments to benefit from this modification.

A critical aspect is the duration of the timeout, called *itsGnNeighbourFlagExpiry* hereafter. If it is too short, a station may treat one-hop neighbours with a low packet rate as remote neighbours accidentally. This behaviour would eliminate them as potential forwarders from GF. Longer *itsGnNeighbourFlagExpiry* durations increase the delay until a more-than-one-hop neighbour is marked as such. Considering the fact that any GN station is required to transmit a beacon every 3.0 s to 3.75 s if no other SHB packets have been sent meanwhile, $itsGnNeighbourFlagExpiry = 3.75\,\text{s}$ is an appropriate compromise for all stations. Of course, vehicles will also send at least one SHB packet per second because of their CA service. On the other hand, RSUs do not send out CAMs as they are not traffic obstacles others should be aware of. The described mechanism could be further extended by using individual *itsGnNeighbourFlagExpiry* settings for each station type. One of 16 available station types is encoded in the GN address each station possesses. The proposed value of 3.75 s remains a reasonable upper bound for any station type, though.

Simulation

It would be possible to forge a scenario with an RSU, which wants to disseminate a warning to more a distant area so that non-area forwarding is necessary at the beginning. The RSU may then select an out-of-range vehicle as first GF hop in the unmodified case and a within-range vehicle in the enhanced case. Honestly, the mere introduction of *itsGnNeighbourFlagExpiry* does not solve the described GF problem entirely but only mitigate it. In some situations, the expired *isNeighbour* may be sufficient to restore successful forwarding. However, it remains an inherent problem of GF that it cannot recover from the selection of a non-reachable station, even it has just out of range shortly before.

Nevertheless, the effect of *itsGnNeighbourFlagExpiry* can still be demonstrated easily. The histograms in Fig. 4.2 show the distribution of neighbour counts in vehicles' LocTs at an arbitrary, common time point. The employed DekiNet2 scenario has 2344 running vehicles at this time. Before taking snap-

shots from LocTs, the vehicles have been exchanging GN beacons already for a minute. As vehicles had time do depart from each other, connectivity of some got interrupted over time. It is then advisable to reset *isNeighbour* for these stations. Though vehicle traffic is identical across all three plots, vehicles are equipped with ITS-G5 at varying rates. This allows to investigate the impact of connectivity patterns but avoids side-effects introduced by different movement patterns. The quota of equipped vehicles starts with 10 %, over to half of all vehicles, up to all vehicles. Each quota's chart brings both *isNeighbour* modes face to face, standard's persistent and the proposed soft-state lifetime of the neighbour flag.

Two observations are remarkable, which apply to all three equipment rates. First, persistent *isNeighbour* flags lead to an overestimation of direct neighbours, which can be seen by the tail to the right end of the histograms. This observation is more pronounced with growing quotas. Second, more stations claim to have no neighbours at all using the soft-state variant. Since having no neighbours is an essential criterion for SCF, this forwarding feature can come into effect more often. Specific enhancements concerning SCF are outlined in Section 4.1.4. Especially at higher quotas, the difference in tracked neighbours grows as shown by the diverging Kernel Density Estimation (*KDE*) lines. Suppose congestion control algorithms consult the number of neighbours as heuristic; In that case, they tend to be misguided in situations where congestion can become an issue, *i.e.* with many equipped vehicles around them.

All in all, soft-state *isNeighbour* flags cannot prevent outdated, unreachable one-hop connections entirely. Yet this approach reduces the lifetime of such stale data in LocTEs significantly from previously at least $itsGnLifetimeLocTE = 20\,\mathrm{s}$ (and possibly infinitely) down to no more than 3.75 s. The implementation effort of the described modifications is almost negligible. Vanetza's implementation stores a timestamp (64-bit integer) instead of a boolean to represent not just that the flag but also when the flag has been set. If the flag has never been set, the stored timestamp equals the minimum representable value.

Figure 4.2: Histograms of tracked neighbours depending on *isNeighbour* lifetime

4.1.3 Glow Forwarding

When a GBC packet is disseminated in the network, this packet is only alive for a limited duration. The upper bound for this duration is the maximum number of hops and the full CBF timer duration for each hop. By default, these are 10 hops and 100 ms resulting in a propagation phase of 1 s maximally [193]. If SCF is enabled, the propagation phase may get suspended until a neighbour becomes reachable. Especially at the border of the link range, connectivity can be unpredictable as slight signal fluctuations affect reception success: Though a packet has been received from a particular station and it is thus known in the location table, a transmission towards this station may still fail. Since vehicles usually send out CAMs every 0.1 s to 1 s, stations are likely aware of other vehicles, though a portion of these messages has not been received. SCF does not even start buffering packets for later dissemination in such a case.

Some information, such as GNSS correctional data, may remain valid for several minutes. Those stations entering the destination area after the propagation phase but before the information becomes invalid, are lacking this information if it is not distributed again. Hence, the DEN service specifies an optional Keep-Alive Forwarding (*KAF*) feature [172] to keep the DENM visible in the network. Also, the GN network entity itself supports packet repetitions. However, each repetition causes to execute all source operations again. Thus, receiving stations cannot distinguish if the packet is a new or a repeated one as they both have a unqiue sequence number assigned in either case. Triggering conditions of some DEN use cases proscribe generation of explicit update DENMs within certain time intervals. Per generated DENM, these triggering conditions also often demand its repetition, which can be realised by GN repetition requests. Updated or repeated DENMs induce the same amount of network load as any newly triggered DENM. Furthermore, the source of dissemination is the same vehicle, which may be comparatively far away from the destination area's fringe.

Since keeping V2X messages alive in sparse networks is a valuable feature, various approaches have been studied before. Reumerman et al. [38] outline the concept of 'message relay boxes'. These boxes are supposed to keep information available in scenarios with low V2X penetration in the market. Their concept intends RSUs to store received messages from vehicles passing by and retransmit them when another vehicle comes into proximity. Though the concept includes databases and controllers to manage those messages, no specific filtering rules and mechanisms are given beyond the conceptual idea. Compared with SCF, described message relay boxes allow more fine-grained control over forwarded messages, because messages may be prioritised and application logic be involved. However, assuming wide-spread deployment of RSUs at low V2X penetration rates is quite optimistic.

Maihöfer et al. [30] compare the performance of three variants to keep a GBC packet alive in its destination area. The first variant requires a permanent, remote server storing all active GBCs in the network and sending them either periodically or on-demand to the destination area. This approach has shown low packet delivery ratios because of the large number of intermediate hops. Another variant elects one station as temporary server responsible for a particular destination area. The locality of this distributed approach enables high packet delivery ratios; however, the election of a single station creates also a single point of failure. The third approach performs equally well by harnessing all neighbours to keep the message alive. Whenever another station enters the destination area, it will be notified by one of the stations already inside. This variant requires each station to track which packets are known to each of its neighbours. However, a GN station can gather this knowledge only by hearing a neighbour sending a particular packet or additional signalling.

Proposal for delay-tolerant Forwarding

Glow Forwarding is a proposed GN extension with the aim to keep the information of a GBC packet alive within its addressed destination area. The principal idea behind Glow Forwarding is to employ those stations reached by the initial CBF propagation as 'hot spots' to keep the packet alive in the network for newly arriving stations. These hot spots repeat the GBC packet at the rate specified by the source station as *Glow Rate*. For this purpose, the Glow Rate is encoded in the GN Extended header using some previously reserved bits. The duration of this repetition procedure is limited by each packet's lifetime. While the dissemination of conventional GN repetitions always starts at the source station, repetitions by Glow Forwarding are spread spatially over the destination area. GBC packets repeated by Glow Forwarding are marked as such by setting the Glow Flag in the GN Basic header. This flag suppresses conventional CBF by receivers and activates the new Glow Forwarding operations. Source stations enable Glow Forwarding by setting a non-zero Glow Rate in their GBC header. Encoding rules for Glow Flag and Glow Rate are detailed later based on Table 4.3 and Fig. 4.3.

Glow Forwarding further relies on a dedicated packet buffer, called 'Glow Buffer' hereafter. Similar to the CBF packet buffer, the Glow Buffer manages timers for each buffered packet. On the expiration of a timer, the respective packet is removed from the buffer and passed on to Glow Forwarding's handler. This handler retransmits the packet and decides on further buffering as outlined by Algorithm 4. Operations supported by the Glow Buffer are summarised in Table 4.2.

At the source station, no modifications with respect to Glow Forwarding are necessary. The rationale is that if the source station is without neighbours, then SCF is the appropriate feature to counteract this issue. If other stations are located nearby, these will execute Glow Forwarding operations in the first place. The source station may then participate as any other station in Glow Forwarding.

Processing of Glow Forwarding at the receiver side is realised as outlined in Algorithm 2. On reception of a GBC packet not marked as being transmitted by Glow Forwarding, the GN protocol operations specified by ETSI are invoked as usual (line 2). The step numbers given in Algorithm 2 correspond to the numbering in [193, section 10.3.11.3]. This path assures the operation of the state-of-the-art forwarding of GBC packets, including CBF, SCF and LocT handling.

Operations specific to Glow Forwarding are only executed if the received packet contains a non-zero Glow Rate. Upon first entrance of this code path, *i.e.* the packet has not been buffered in the local Glow Buffer yet, the packet will be added to the buffer, and the local station begins with Glow Forwarding. Additionally, the packet will be passed to the upper layer if the station has not received this packet before by conventional GBC forwarding such as CBF. If Glow Forwarding is already in progress, *i.e.* Glow Buffer contains the received packet already, the associated timer's expiry might get deferred to avoid duplicate transmissions by nearby stations at similar time points. This deferral is determined by Glow Forwarding's reluctance, a factor in the range from 0.0 to 1.0. As a rule of thumb: The closer the forwarder is to the receiving station, the more the timer will be deferred because of a large reluctance factor. Under no circumstances, however, the timer will expire earlier than scheduled previously, not even due to receptions from very remote forwarders. Otherwise, timer expiry may get truncated mistakenly whenever briefly after a nearby repetition also a far-away repetition is heard.

Similar to GN beacons, jitter is added to Glow Forwarding's time points of transmission. Actual timer expiry depends not only on the glow rate encoded in the packet's GBC header but also some uniformly drawn random jitter. This is meant to reduce the risk of collisions on the radio channel. Without adding random jitter, all concurrent receivers of a packet with enabled Glow Forwarding would schedule their respective retransmissions at equal time points.

Algorithm 3 details the calculation of the reluctance factor when deferring Glow Forwarding timers. The algorithm looks up the position of the forwarder, identified by its MAC address $\mathrm{MAC}_{\mathrm{fwd}}$, in its LocT. If the look-up of a matching LocTE succeeds and contains a valid Position Vector (*PV*), the distance between the forwarder and the receiving station's EPV is calculated. The ratio of this distance to the maximum distance, given by the GN protocol

Algorithm 2 Glow Forwarding for GBC packet P by receiving stations

```
1:  ProcessBasicAndCommonHeader(P)                          👍 steps 1 and 2
2:  if ¬P.GlowFlag then
3:      ExecuteConventionalGbcFwd(P)                        👍 steps 3 to 13
4:      if EPV ∈ P.DestinationArea then
5:          PassUp(P)                                       👍 step 14 (last)
6:      end if
7:  end if
8:  if P.GlowRate > 0 then
9:      if ¬ GlowBuffer.contains(P) then
10:         jitter ← randuniform(0.75, 1.25)
11:         T_start ← P.GlowRate × jitter
12:         GlowBuffer.store(P, T_start)
13:         if P.GlowFlag ∧ EPV ∈ P.DestinationArea then
14:             PassUp(P)          👍 not received by GBC forwarding before
15:         end if
16:     else
17:         reluctance ← CalculateGlowReluctance(MAC_fwd)
18:         jitter ← randuniform(0.25, 0.5 + reluctance)
19:         T_update ← P.GlowRate × jitter
20:         T_expiry ← GlowBuffer.expiry(P)
21:         if T_update > T_expiry then
22:             GlowBuffer.store(P, T_update))   👍 defer Glow Forwarding
23:         end if
24:     end if
25: end if
```

constant itsGnDefaultMaxCommunicationRange, governs the reluctance factor then. In the (theoretical) case of forwarder and ego being located at the same position, a factor of 1.0 is returned. This is also the fallback factor if the distance cannot be calculated for some reason. Though a factor of 0.0 may be returned for forwarders far away, this does not cause an immediate retransmission because of the $T_{update} > T_{expiry}$ condition in line 21.

Algorithm 3 Calculate reluctance of Glow Forwarding (continuation of Algorithm 2)

```
26: function CalculateGlowReluctance(MAC_fwd)
27:     if ∃LocTE ∈ LocT : LocTE.MAC = MAC_fwd ∧ LocTE.hasPV then
28:         d_fwd ← distance(EPV, LocTE.PV)
29:     else
30:         d_fwd ← 0
31:     end if
32:     d_max ← itsGnDefaultMaxCommunicationRange
33:     return max{d_max − d_fwd, 0.0}/d_max
34: end function
```

Actions taken by Glow Forwarding upon timer expiry of a buffered packet are outlined in Algorithm 4. The Glow Buffer invokes this handler and passes the packet and the buffering duration $T_{buffered}$, *i.e.* the time span

from initially inserting the packet until now. Only if this time span does not exceed the packet's lifetime and the current ego position EPV is still within the addressed destination area, Glow Forwarding actually retransmits the packets. The retransmission decision concerning the destination area is deliberately postponed to the timer expiry because the station may have moved considerably since the time point of reception, especially for long glow rate cycles. Before actual retransmission, packet's Glow Flag is set and its lifetime reduced by the queuing time. As a safety measure against broadcasting storms by legacy stations not aware of Glow Forwarding, the 'Remaining Hop Limit' counter of the packet is zeroed. After transmission, the very same packet is rescheduled in the Glow Buffer. Otherwise, when the packet itself has expired, or the station has left the destination area, the packet is discarded. Glow Forwarding stops at this point by not starting another timer for this packet in the Glow Buffer.

Algorithm 4 Glow Forwarding handler
invoked by Glow Buffer on timer expiry

```
35: function OnGlowTimerExpiry(P, T_buffered)
36:     if P.Lifetime > T_buffered ∧ EPV ∈ P.DestinationArea then
37:         P.GlowFlag ← true
38:         P.Lifetime ← P.Lifetime − T_buffered
39:         P.RemainingHopLimit ← 0                    👉 safety measure
40:         PassDown(P, MAC_broadcast)                 👉 transmit P on link layer
41:
42:         jitter ← randuniform(0.75, 1.25)
43:         T_expiry ← P.GlowRate × jitter
44:         return T_expiry                👉 reschedule P, continue Glow Forwarding
45:     else
46:         return 0                                   👉 discard P, stop Glow Forwarding
47:     end if
48: end function
49: GlowBuffer.handler ← OnGlowTimerExpiry
```

Previously unused bits in the GBC header can be used to encode Glow Flag and Glow Rate, as shown in Fig. 4.3. Since Glow Flag needs to be settable by any forwarding station, this flag needs to get placed in the GN Basic header as the secured message's signature does not cover this header. Fortunately, a 'reserved' field of 1 B length exists that can easily accommodate the Glow Flag. Glow Rate shall only be governed by source stations and thus placed in a signed header section. From the two options, Common and Extended header, the Extended is chosen because Glow Forwarding is only reasonable for packets addressing geographic areas. While a Common header is part of every GN packet, the particular Extended header depends on the GN transport type. Again, the Extended header for GBC transports contains a yet unused, reserved field of 2 B. Table 4.3 proposes a 4-bit encoding of Glow Rate ranging from 0.5 s to 300 s. A minimum glow rate of 0.5 s seems

Table 4.2: Operations on Glow Forwarding packet buffer 'GlowBuffer'

Operation	Effect
`store(P, TO)`	stores packet `P` for duration `TO`; resets timer expiry to `TO` if packet is already buffered
`contains(P): bool`	check if packet `P` is already buffered
`expiry(P): TO`	duration from now until timer of packet `P` expires; returns zero if packet is not buffered
`<handler>(P, T): TO`	invoked on timer expiry of packet `P`; `T` is the total duration `P` has been buffered en bloc; return value `TO` affects further buffering: TO $>$ 0: reschedule timer of `P` with expiry `TO` TO $=$ 0: discard packet `P`

Table 4.3: Encoding of Glow Forwarding rate as 4 bit wide header field

Field Bits				Glow Rate $^{-1}$
0	0	0	0	disabled
0	0	0	1	0.5 s
0	0	1	0	1 s
0	0	1	1	2 s
0	1	0	0	3 s
0	1	0	1	5 s
0	1	1	0	7 s
0	1	1	1	10 s
1	0	0	0	15 s
1	0	0	1	20 s
1	0	1	0	30 s
1	0	1	1	1 min
1	1	0	0	2 min
1	1	0	1	3 min
1	1	1	0	4 min
1	1	1	1	$0.5 * \mathrm{LT}_{max} \approx 300\,\mathrm{s} = 5\,\mathrm{min}$

appropriate even for high-speed vehicles with $v = 70\,\mathrm{m/s}$. In the worst case, such a vehicle would enter the destination area by not more than 35 m before receiving a glow forwarded packet. At most, roughly half of the maximum packet lifetime can be encoded, which is $\mathrm{LT}_{max} = 630\,\mathrm{s}$ [193, section 9.6.4]. Considering the lifetime check in Algorithm 4, a Glow Rate of almost LT_{max} is the upper bound where Glow Forwarding can become effective at all but without limited benefit with respect to the remaining packet lifetime.

Proof-of-Concept Simulation

Results of a preliminary evaluation of Glow Forwarding are shown in Fig. 4.5 and Fig. 4.4. These figures are based on simulation results employing an excerpt of DekiNet2 with a rural intersection. At this intersection, an RSU is transmitting a GBC packet with a circular destination area of 1 km radius

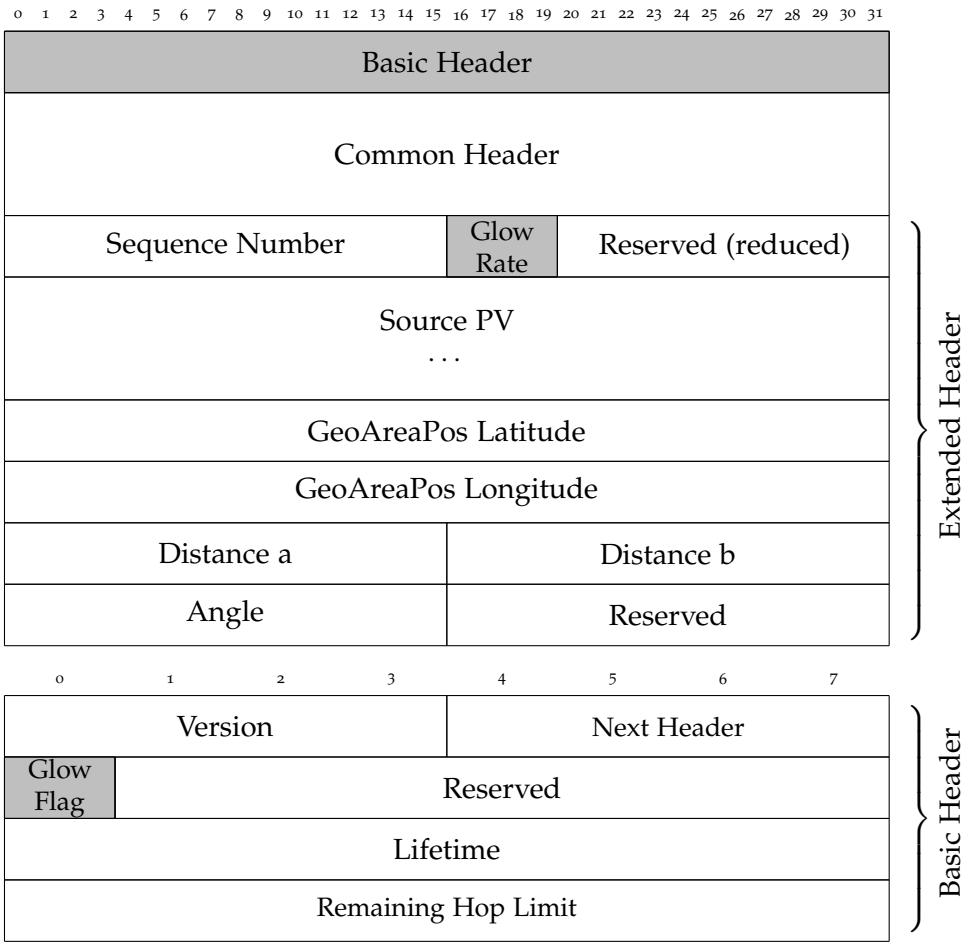


Figure 4.3: GBC header adapted for Glow Forwarding

centred at its position. In the baseline simulation, the RSU repeats this packet every 5 s for 30 s, *i.e.* the network is confronted with seven individual CBF phases. These seven source transmissions are accompanied by 44 forwarding transmissions, *i.e.* 51 transmissions in total. All 19 vehicles within the destination area receive the packet's information with this repeated CBF pattern over time. However, the initial CBF phase reaches only 10 vehicles, *i.e.* repetition is vital for good coverage.

The Glow Forwarding variant does not repeat the GBC requests but employs a glow rate of 5 s instead. The traffic in the otherwise identical network consists of 1 source transmission, 6 CBF transmissions and later on 43 Glow Forwarding transmissions, summing up to 50 transmissions. One vehicle within the destination area is missed, though, which may be just unfortunate in this particular situation. Final conclusions regarding the performance of both variants cannot be drawn yet based on a single simulation scenario. Representative simulations and evaluations in Chapter 5 will address forwarding performance in detail.

Nevertheless, it can be noticed that Glow Forwarding causes roughly the same amount of transmissions like conventional CBF with a comparable repetition rate. As the histograms in Fig. 4.4 show, channel access is actually

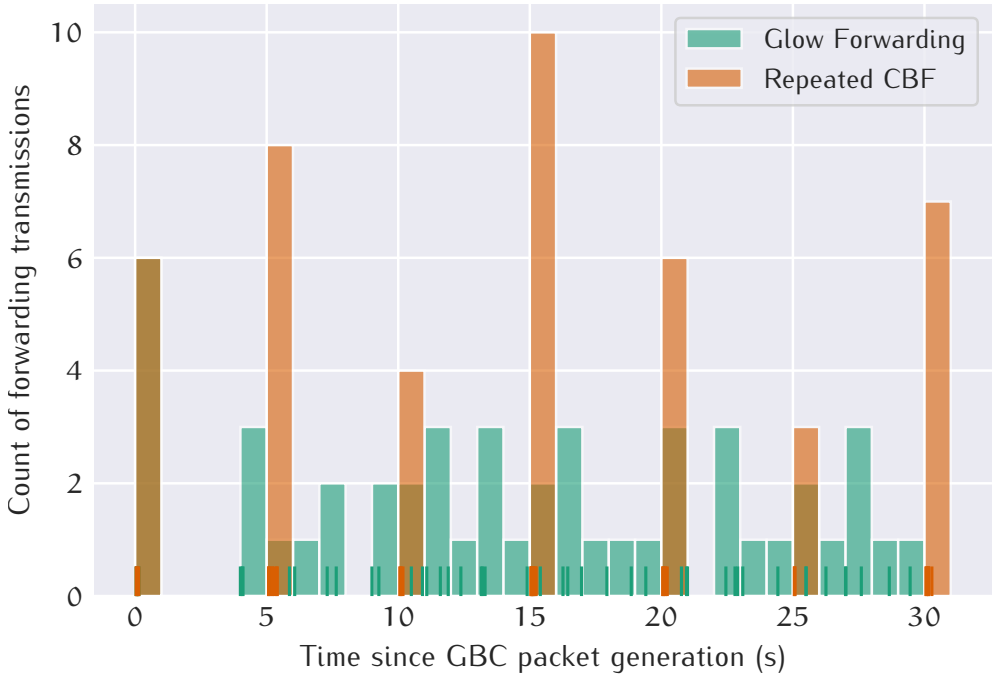
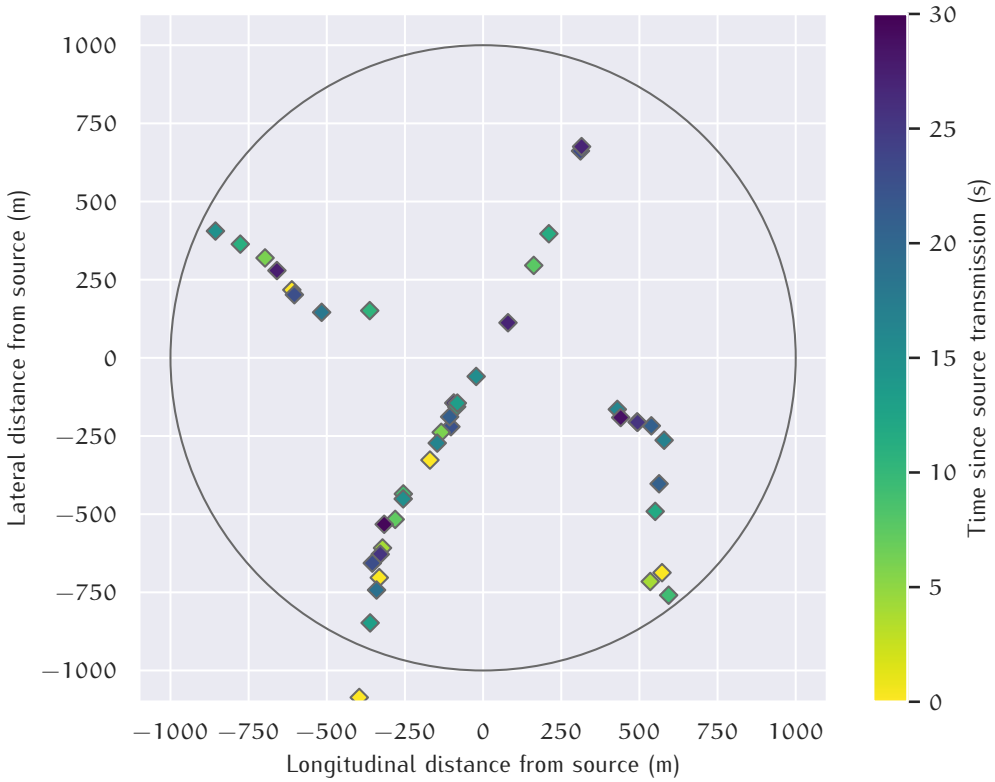


Figure 4.4: Forwarding time points of Glow Forwarding versus repeated CBF

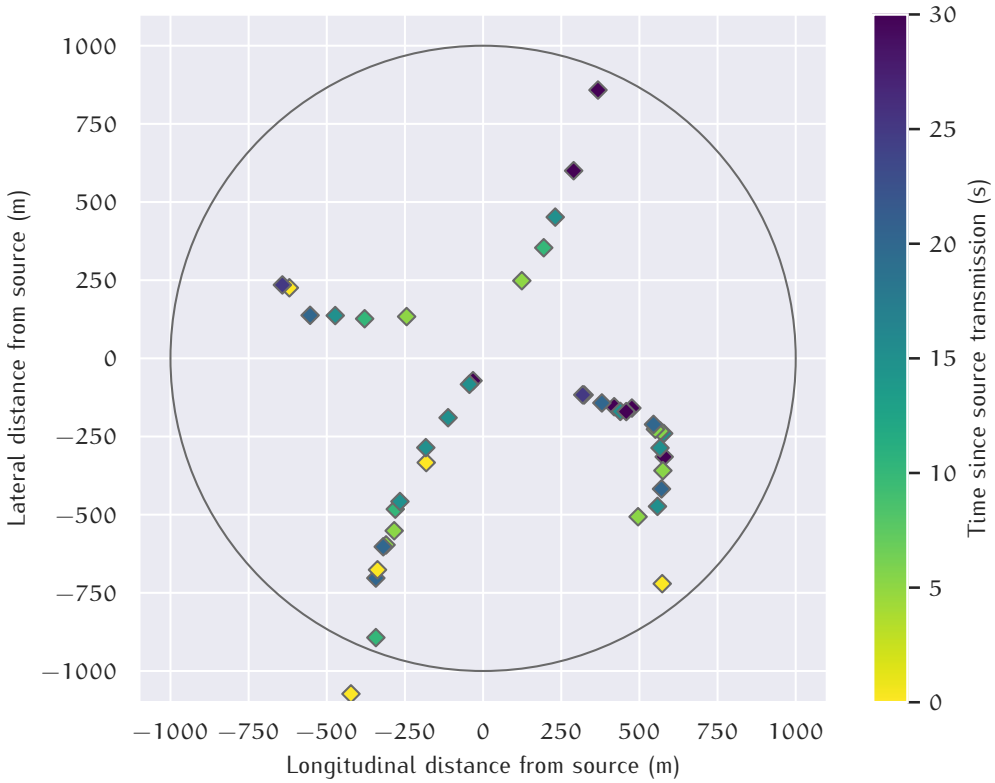
spread over time using Glow Forwarding, while CBF transmissions cluster after each RSU transmission. The time-dependent colouring of markers in Fig. 4.5 also reveals that forwarders exist on each intersection leg over the whole packet lifetime of 30 s. Hence, the information remains indeed 'alive' within the destination area as desired.

Two further comments regarding Fig. 4.5: First, random jitter has been applied to the positions in Fig. 4.5 to avoid visual problems caused by over-plotting. Equal positions in raw data may thus be plotted slightly off within each sub-figure and also across them. Second, the forwarding station outside the destination area in the southwest is not an error. If an outside station cannot determine where the previous sender is located, non-area forwarding is selected as the safe fallback procedure [193, Annex D]. This constellation quickly occurs if no earlier SHB packet has been received from the sending station, *e.g.* both stations have just come within communication range. The receiving station has no matching LocTE yet to lookup the sender's position based on its MAC address.

Last but not least, Glow Forwarding opens up long-term scheduling of packets. While each CBF repetition has a unique sequence number, this identifier remains fixed during Glow Forwarding. Thus, a component at the network layer could observe occurrences of equal payloads easily by only comparing rather short identifiers instead of the byte-wise comparison of full payloads.



(a) Glow Forwarding



(b) Repeated CBF

Figure 4.5: Location of forwarders using Glow Forwarding versus repeated CBF

4.1.4 Handling of Store-Carry-Forwarding by Forwarders

Forwarding relies on third-party stations to act as relays for transporting information from their source to a destination station (GeoUnicast (*GUC*)) or destination area (GBC). In networks with a low density of ITS-G5 stations, *e.g.* in the ramp-up phase of deployment or in rural environments, forwarding may get stuck as no suitable forwarder is available at the moment. For delay-tolerant packets, one can mitigate this issue by temporarily buffering the packet until an appropriate forwarder gets into reach. This method is called SCF in GN, and its function principle has been outlined in Section 2.2.2.

The GN specification says that a packet is enqueued in the respective SCF buffer if the SCF flag is set and the LocT contains no entry marked as a direct neighbour. This behaviour is acceptable for any packet generated by the local station, *i.e.* as part of source operations of SHB and GBC requests invoked by upper layers. However, this behaviour has its shortcomings during forwarding operations of GBC packets: When a station receives a GBC packet, it likely has a direct neighbour in its LocT, namely the sender of the very same packet. Consequently, the receiver will forward immediately using CBF, for example, but the packet effectively only 'bounces back' to the previous sender. The original goal of increasing the coverage range by tolerating some dissemination delay is defeated then.

Handling of SCF by forwarding stations can be divided into two aspects. First, on the reception of an SCF-enabled packet, the station has to decide either to proceed with area- or non-area forwarding immediately or to *store* it in the SCF buffer temporarily. Second, on the reception of a subsequent packet the station needs to determine if this sender is suitable to *carry on* previously buffered SCF packets.

The aspect of dealing with the initial buffering decision is called the *buffering policy* in the following. The selection procedure which buffered packets to keep and which to forward now when flushing the buffer is called *flush policy*. To clearly distinguish between the original sender, whose packet got buffered, and the sender of a subsequent packet, former is called 'source' and latter 'candidate'. The *flush policy* thus checks if 'candidate' is a suitable next hop for any buffered packet from 'source'.

Table 4.4 lists up the respective policies for the standardised behaviour and two proposed alternatives. The standardised behaviour does not distinguish between source operations for newly generated packets and forwarding operations for multi-hop dissemination. Standard's simple policies are reasonable for source operations: buffer if no neighbours are known and flush all packets unconditionally on the reception of the first subsequent packet. As stated at the beginning, a station will rarely find no LocTE of neighbours during forwarding operations, which makes this *buffering policy* ineffective.

Table 4.4: Buffering and Flush Policies of Forwarders for SCF

Variant	Buffering Policy	Flush Policy
Standard	no neighbours are present in LocT	unconditionally all packets
Skip Sender	no neighbour except sender is present in LocT	all packets not originally received from candidate
Check Topology	no neighbours are known in opposite direction than sender	all packets received from opposite direction than candidate

The 'Skip Sender' alternative adds MAC address checks. A packet will be buffered exclusively if the sender is the only known neighbour. The other way around, a packet is only flushed if the candidate's MAC address is different from the source's address. Otherwise, the buffering station will wait until a suitable carrier comes into reach.

With 'Check Topology' the current network topology as known through the LocT is evaluated. A packet is buffered if all other stations known as neighbours are located in the same direction as the source station. Two stations are considered to be in the same direction if their bearings from the receiver's position differ by less than 180°. Vice versa, only those packets are flushed which have been previously received from the opposite direction. Geometrically speaking, the receiver cuts the plane into two half-planes where the cut edge is the perpendicular to the source's position. If the source and the candidate are not located in the same half-plane, they are considered to be in opposite directions.

The GN standard tells to flush the SCF buffer before the *Extended* header is processed, *i.e.* also before updates of LocTEs when an SHB packet or GN beacon has been received. With the more sophisticated flush policies in place, the invocation to flush this buffer is deferred until pending LocTEs updates have been applied. This slightly changed control flow prevents false decisions because of outdated data.

The concept of all three variants has been checked using the Luxembourg scenario at night. The vehicle density is low at night hours, and thus connectivity is interrupted from time to time. One vehicle is generating a GBC packet addressed to all 17 vehicles within 1 km distance. The effect on the dissemination of each variant is shown as a *latency map* in Fig. 4.6. Each marker denotes the position where a vehicle has received the packet for the first time. The marker colour encodes the latency measured from the time point of packet generation by the source vehicle. For comparison with SCF, the plot in the bottom right show the performance of Glow Forwarding with a glow rate of 3 s. Table 4.5 supplements these plots by giving numbers describing the impact of each variant.

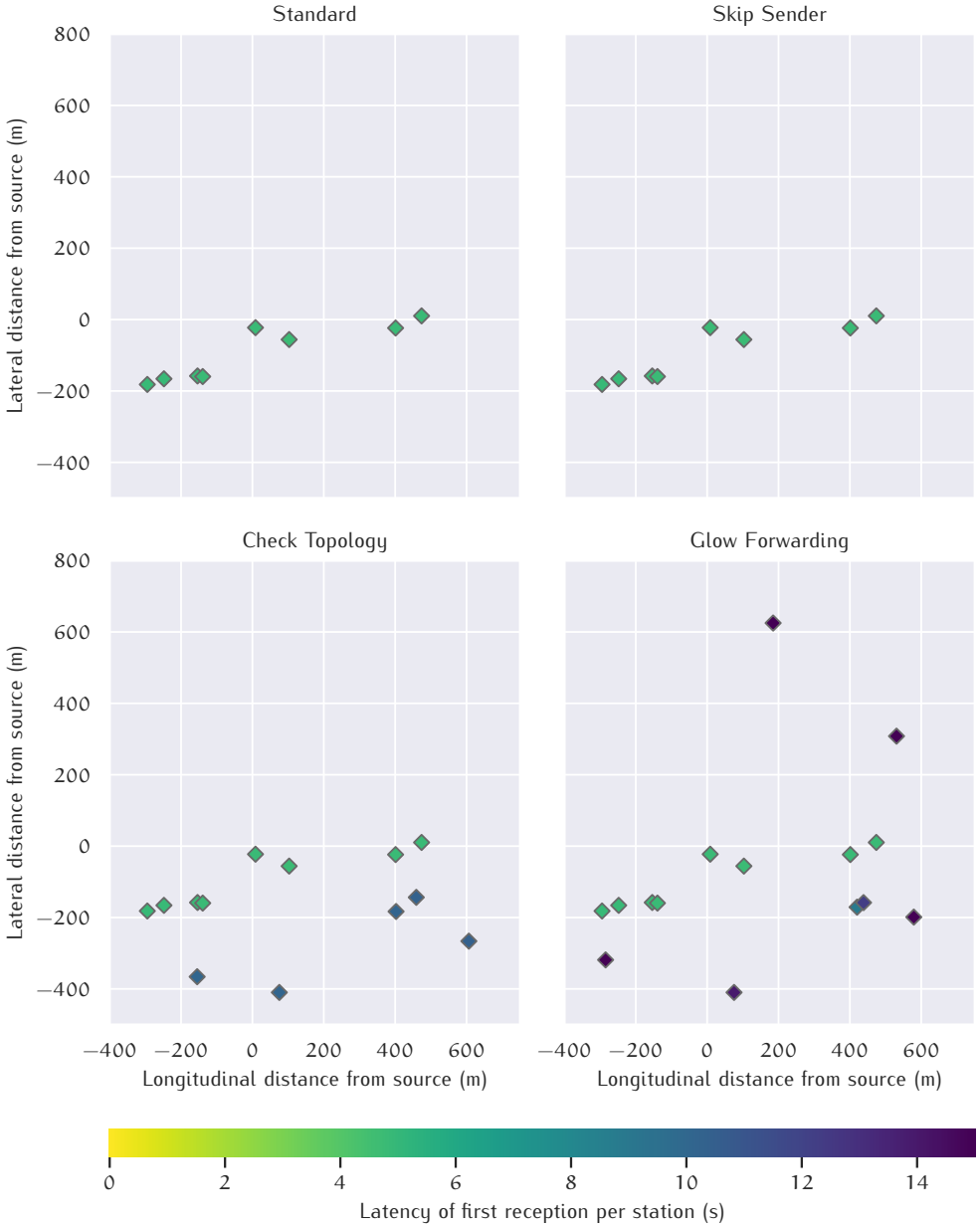


Figure 4.6: Latency map plots of proposed SCF policies

Table 4.5: Result of SCF variants in numbers

Variant	Reached (unique vehicles)	SCF Stores	SCF Carries	Forwardings (in total)
Standard	8	1	1	5
Skip Sender	8	2	2	6
Check Topology	13	14	6	24
Glow Forwarding	15	–	–	48

As expected, the standard SCF variant buffers reluctantly only once. Plain CBF area-forwarding reaches 8 out 17 vehicles then. Unfortunately, the *Skip Sender* policy does not show any improvement in terms of reached vehicles. At least twice a packet is actually put in an SCF buffer ('store'), though, and flushed latter on ('carry'). Coverage is significantly increased by *Check Topology* policies, reaching 13 of the 17 addressed vehicles. It seems to pay off that this policy stores packets more often than the prior policies. Notably, *Check Topology* is also more eager to keep buffered packet as only 6 out of 14 are later transmitted.

Glow Forwarding reaches roughly the same vehicles like the *Check Topology* policies. Latencies observed by vehicles, which are not reachable from the beginning, tend to be higher because Glow Forwarding does not trigger immediate transmissions upon reception. The two more distant vehicles in the northern part can be reached by Glow Forwarding but not SCF, though. Glow Forwarding comes with the cost of twice as many forwarding transmissions compared to *Check Topology*. However, it is not justified to stamp Glow Forwarding as too chatty right off. Glow Forwarding is designed to extend retransmission timers of buffered packets when duplicates are received. In a sparse network, as the one studied here, Glow Forwarding may just repeat packets into the void. Induced channel load is thus irrelevant if no other network participants are nearby.

In conclusion, SCF with *Check Topology* and Glow Forwarding are able to bridge sparse connectivity, while standardised SCF is not. Which of both is more effective (latency and coverage) and efficient (number of transmissions), will need to be studied using a more diverse set of scenarios.

4.1.5 Cancellation of Long-Term Packets

Among Day One ITS applications, the DEN service is the textbook example of a service with long-term communications. Besides messages at detection of events, this services also generates messages to update or negate prior messages or even terminate events. The DEN service even repeats DENMs with unmodified payload at the application level, though a packet repetition feature is also exposed via GN's *GN-DATA.request* interface to upper layers [193]. Unfortunately, standardisation does not foresee any interface to cancel repetitions handed over to GN. If the DEN service generates a DENM update, the same GN entity may thus still pass on outdated DENM repetitions intermixed with updated DENMs. From this point of view, it is understandable that the DEN service is eager to avoid such unfavourable situations and opts to manage repetitions on its own. However, DCC could benefit from the knowledge about pending repetitions for scheduling channel resources ahead of time.

Extended GN Interface

Upper layers can set two parameters when passing a packet for transmission via *GN-DATA.request*, which control packet repetition by GN: The repetition interval and the maximum repetition time. The corresponding *GN-DATA.confirm* conveys only a result code indicating success or an error condition according to current GN specification. Please consult ETSI [193, Annex J] for a full description of the *GN-DATA* service primitives.

This interface gets enhanced by an optional *repetition handle*, which is set if the repetition parameters have been set in the preceding *GN-DATA.request*. This handle identifies the particular repetition procedure scheduled by the GN entity. If the triggering service, *e.g.* the DEN service, intends to cancel a repetition under certain circumstances, it has to remember this handle.

Upper layers can then cancel each running repetition individually by notifying GN via the novel *GN-DATA.cancel*. The *repetition handle* is the only parameter conveyed by this interface from an upper layer to GN. If GN still finds a running repetition for the given handle, further repetitions are discarded immediately. If the handle matches no running repetition, *e.g.* its maximum repetition time has already been reached, no action needs to be taken by GN.

The implementation effort for this proposed enhancement is low as the reference implementation in Vanetza shows. Vanetza realises the *repetition handle* as plain `uint64_t`, which is stored by GN along with the request parameters and the payload. Latter data had to be stored already to realise GN repetitions at all. Thus, memory requirements increase moderately by a single integer per to-be-repeated data request. Passing *repetition handles* back and forth via the extended interfaces is also an easy problem to solve.

Cancellation of Glow Forwarding

With the introduction of Glow Forwarding in Section 4.1.3, the accompanying Glow Buffer also stores packets for long-term repetitions. In contrast to classical GN repetitions, the time pattern of repetitions by Glow Forwarding is not predetermined. Nevertheless, the same motivation for sooner cancellation applies to Glow Forwarding.

Similar to the approach outlined for classical GN repetitions, a *Glow Handle* can be passed between GN and upper layers. However, it is not sufficient to remove the associated packet from the Glow Buffer only. Since neighbouring stations will still transmit this packet via Glow Forwarding, the local station would simply re-add this packet to its buffer on next reception.

Keeping books of (locally) cancelled Glow Forwarding packets in a 'suppression list' can prevented these re-additions. When GN is asked to cancel Glow Forwarding, it shall remove the packet from the Glow Buffer and add

the corresponding packet identifier to this list. These list entries have a lifetime equalling the remaining lifetime of the previously buffered packet. Line 8 of Algorithm 2 is amended by a check if the respective packet is enlisted as per Algorithm 5. If this is the case then further Glow Forwarding steps are skipped, and thus participation of the local station in forwarding of the cancelled packet is suppressed. Vanetza's reference implementation of the proposed mechanism employs the tuple of GBC source address and sequence number as handle and packet identifier likewise.

Algorithm 5 Suppression of cancelled Glow Forwarding (modification of Algorithm 2)

```
1: if P.GlowRate > 0 ∧ ¬SuppressionList.contains(P) then
2:     Further steps as in Algorithm 2
3: end if
```

In conclusion, the benefit of packet cancellation at GN layer is not only the reduced number of transmissions but to allow for resource planning by DCC. If individual services implement repetition mechanisms on their own, it becomes tedious for DCC to estimate the upcoming demand by packet repetitions. Gathering this information from the GN entity is comparatively easy, on the other side.

4.2 ENHANCEMENTS FOR DECENTRALIZED CONGESTION CONTROL

The motivation to enhance the GN protocol is to avoid unnecessary or untimely transmissions in the first place. DCC's task is it to advice upper layers about suitable transmission time points and manage channel usage appropriately – internally among competing services and externally among stations.

Transmission Demand Analysis

Considering the architecture of ITS-G5 and its use cases, five major groups emerge a transmission belongs to:

1. *Emergencies* Messages triggered in case of an emergency are generated with the highest DP. Following the treatment of emergency messages by BSP [163], a dedicated emergency budget on top of the generally assigned channel capacity is employed. Even if this budget causes channel usage exceeding limits such as the CBR target temporarily, this is deemed acceptable if danger is averted.

2. *Advisory Events* Another group of DENMs is generated not because of emergency but to inform drivers about less critical situations, *e.g.* road hazards or traffic jams. These event-triggered messages have to be treated in the regular channel access budget, but their occurrence is hard to predict. If such a message arrives at the link layer, it will preempt periodic messages of lower priorities.

3. *Periodic Packets* With CA and CP exist two periodic services, which explicitly take DCC's TRC constraints into account. A novel DCC interface presented in Section 4.2.3 allows periodic services to announce their desired packet rates and give feedback about their current channel usage intent. The semantics of this interface guarantee low-latency dissemination to the respective service. As updated messages are continuously generated, their importance is considered not as high as event-triggered messages.

4. *Repetitions* The addition of the cancellation feature introduced in Section 4.1.5 makes GN repetitions an alternative to repetitions purely controlled by services. From DCC perspective, it is convenient if timing and number of repetitions can be looked up at a single entity, the GN router. Since repetitions are not as urgent as entirely new packets, their repetition time points do not need to be exact. However, the priority in terms of DP of repetitions remains unchanged to the first occurrence of the respective packet.

5. *Forwarding* Up to now, forwarding is not considered specifically by DCC. As a result, a DCC gatekeeper unaware of forwarding breaks CBF [200]. In this thesis, an alternative approach in treating forwarding traffic is outlined in Section 4.2.4. Besides preserving the overhearing of CBF, this alternative also aims to prevent starvation of forwarding packets.

The amount of locally generated packets can be estimated because the upper rate of periodic services is known, and it is unlikely that a single vehicle detects a vast number of events instantly. However, such an upper rate likely overestimates the actual demand as message length and rate can vary significantly. This is especially true for highly adaptive periodic service such as CA. Employing an overestimated demand as the foundation for planning channel usage would lead to subpar channel usage.

On the other hand, the group of forwarding packets – CBF, SCF and the new Glow Forwarding – has been largely ignored in prior work. If estimating the actual channel usage by locally generated packets is already tricky, anticipating the amount of to-be-forwarded packets is even more challenging as it depends on the event detection of all vehicles in a radius of hundreds of meters.

Objectives of Congestion Control

Of course, the main objective of DCC is to keep the channel congestion within defined bounds. On top of this main objective, *TR 101 612* [96], an ETSI report on DCC's performance, mentions the following performance characteristic expected from DCC:

- resilience, *i.e.* DCC achieves sufficient level of QoS, *e.g.* not too conservative and not overly congested
- scalability, *i.e.* DCC keeps performance level to increasing number of vehicles quickly
- responsiveness, *i.e.* DCC adapts parameters fast enough to maintain performance under brief, sudden, recurring changes of channel usage
- adaptability, *i.e.* DCC adapts on changing input conditions

With these favourable characteristics in mind, several changes of DCC components are proposed in this section. At first, the sharing of CBR measurements is revisited by the inclusion of directional information, which enables to localise the shared CBRs. This enhancement contributes to an ITS-G5 station's perception of the prevailing congestion in its surroundings.

Based on the perceived congestion level, mechanisms such as reactive or adaptive DCC as discussed in Section 2.1.4, determine the station's allowed channel usage. Channel usage is given as the maximum duration a station is entitled to occupy the channel for transmissions, also called 'duty cycle'. While the adaptive DCC algorithm explicitly determines this duty cycle, the state-machines of reactive DCC give this duration implicitly as the product of the determined packet rate and the given maximum packet duration.

Further enhancements proposed in this section deal then how to split up the available channel capacity given by the duty cycle among the station's various entities willing to consume this capacity. An enhanced DCC access controller is introduced in Section 4.2.5, which is aware of ITS-G5 specific traffic types demanding for channel resources. In contrast to the existing concept of a gatekeeper at DCC_ACC, this controller's purpose is not only to protect against excessive transmission requests by upper layers but also to coordinate transmissions among those entities. The result is an overall enhanced DCC entity, which guarantees bounded processing of periodic messages (see Section 4.2.3), collaborates with GN to drain SCF buffers gradually (see Section 4.2.2), and handles forwarding packet dedicatedly.

4.2.1 Directional Reporting of Channel Busy Ratios

Measuring the CBR is the most important indicator for the level of channel congestion. The local measurements, *i.e.* those CBRs provided by the station's own MAC layer, represent only the limited perspective of channel congestion at the station's current position. It is hard to predict the near-term evolution of channel congestion as the shadowing of radio waves may change abruptly. For example, a vehicle may get exposed to significantly more radio traffic when entering an intersection as cross-traffic becomes receivable.

Previous publications and current standardisation thus enhance the local measurements by CBR provided by the network. For this purpose, each station encodes its local CBR and the maximum CBR reported by its neighbouring stations in the GN packet header of SHB transmissions. Hence, the horizon of a station to determine channel congestion is extended by two hops. DCC algorithms then react according to the maximum CBR measured locally, reported by its neighbours (one-hop CBR) and neighbours' neighbour (two-hop CBR). Obviously, this approach is quite pessimistic as it only considers the highest CBR of all.

This section aims to develop an alternative DCC information-sharing scheme, which reveals from where the shared CBRs are coming. For this purpose, directional information is added to the one-hop CBR encoded in the SHB headers. The header design and processing mechanisms associated with directional CBRs are the main topics in this section. Later in Chapter 5, the implications of this novel DCC information scheme are studied and how stations can benefit from it.

Header Design

Figure 4.7 presents the DCC Multi-Channel Operations (*DCC-MCO*) field, which is part of every ITS-G5 SHB header. Since this field is an ITS-G5 specific extension of GN, its definition is found in the media-dependent extension of GN *TS 102 636-4-2* [195]. As has been outlined in Section 2.4.3, the information-sharing scheme is inspired by the prior work on PULSAR by Tielert [100]. In particular, the local $\mathrm{CBR}_{\mathrm{L}}^{0}$ and maximum CBR of one-hop neighbours $\mathrm{CBR}_{\mathrm{L}}^{1}$ encoded in the DCC-MCO field are relevant for information sharing. For each of these CBR sub-fields, 8 bit are available with a linear encoding, *i.e.* the 8-bit integer encodes the floating-point CBR in discrete steps of $1.0/255 \approx 0.4\,\%$.

An alternative to the standardised DCC-MCO field should not consume more space and thus fit into the same 4 B slot of the SHB header. Which variant is actually encoded in a particular SHB packet could be noted in the version field available of the *Basic* GN header to ensure backwards compatibility. On this basis, Fig. 4.8 is the proposal named 'DCC-MAP' to

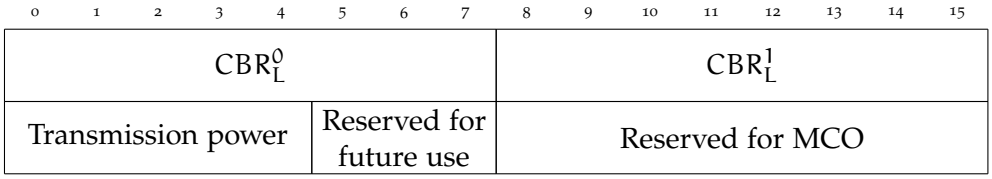


Figure 4.7: DCC-MCO field as defined by ETSI [195]

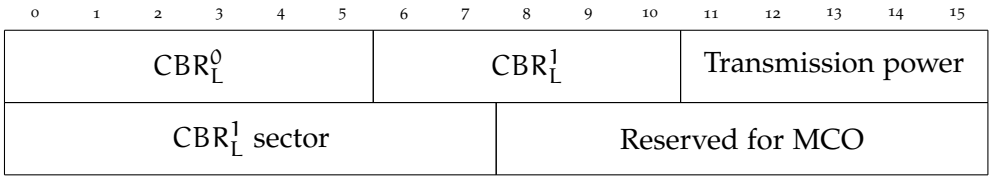


Figure 4.8: Proposed DCC-MAP field for directional CBR reports

support directional CBR reports. The absolute, geographic position belonging to CBR_L^0 is known from the source position vector included in the SHB header. Previously, CBR_L^1 has been the mere maximum of CBRs reported by one-hop neighbours and thus lacking a sense of direction. This flaw is eradicated by the newly introduced 'CBR_L^1 sector' field, which encodes the distances and direction of the included CBR_L^1 measurement.

The length of the CBR fields has been reduced on the cost of the accuracy of their encoded measurements. Otherwise, the header could not accommodate the additional 8 bit wide field. The impact of this reduced accuracy is mitigated by non-linear encoding, *i.e.* low CBR measurements are encoded with larger discretisation errors. As long as the channel congestion is low, DCC regulates stations only mildly if at all. Thus, coarse information sharing about neighbouring CBRs is acceptable. Referring to *TS 102 687* [143], reactive DCC remains in 'relaxed' state as long as CBR remains below 30 % and the target CBR of adaptive DCC is at 68 %.

To get most out of the available bits, the local CBR_L^0 is encoded in relatively coarse 2.5 % steps up to 30 %. Above 30 %, the encoding increases in 1 % steps. CBR measurements exceeding 81 %, are all encoded with the highest possible integer value 63. However, such a heavily congested channel should never occur with DCC at all.

Only 5 bit are available to convey the one-hop CBR measurements CBR_L^1. Similar to CBR_L^0, the encoding of measurements below 30 % has a low resolution, in this case of only 5 %. Between 30 % to 80 %, the measurement is encoded in 2 % steps and is again capped above this upper limit. How the reduced accuracy of the encoded CBRs affects the information sharing is going to be studied later in this section.

The distance and direction of the CBR_L^1 is given relative to the absolute position of the sender. With the sender's position at the centre, Fig. 4.9 visualises the employed encoding scheme. In essence, the distance is given in terms of concentric rings and the direction as a sector on a particular ring. The outer arc length of each ring sector is about 80 m, and thus the outermost

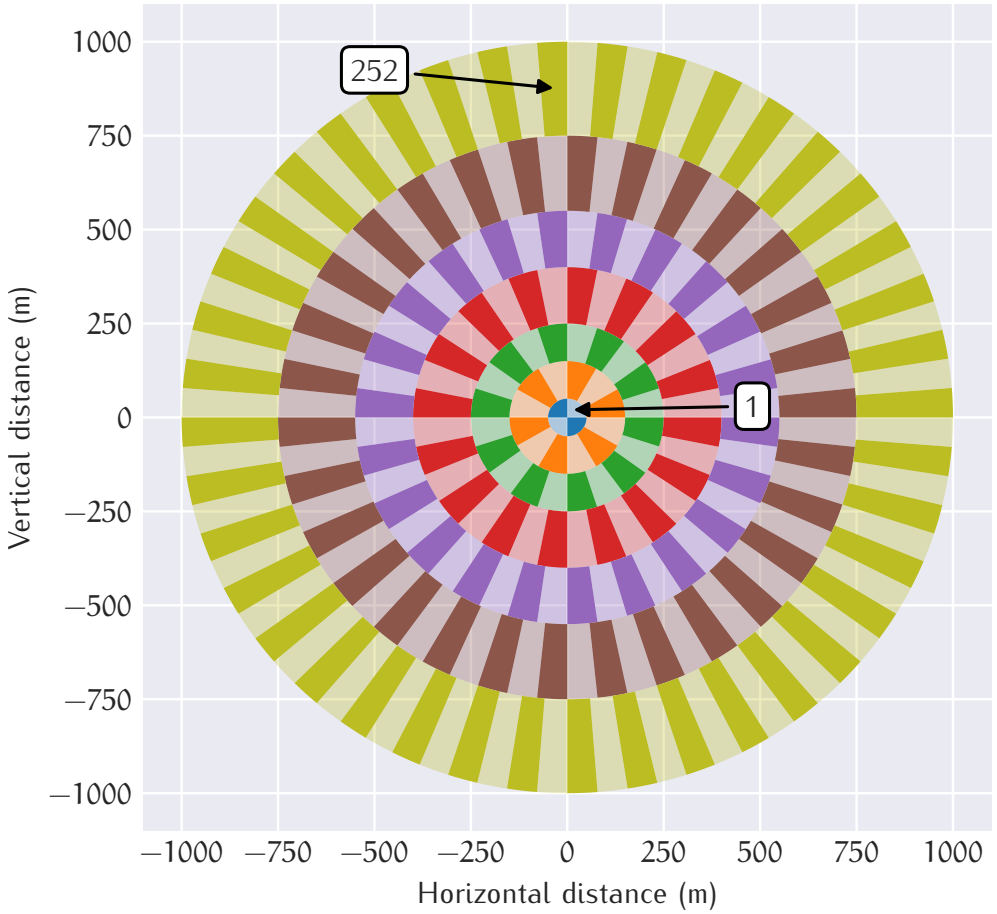


Figure 4.9: Arrangement of 252 ring sectors to encode origin of one-hop CBRs

ring for distances between 750 m to 1000 m consists of 80 sectors. Table 4.6 lists the distance ranges and number of sectors of each ring. Since one-hop CBR_L^1 measurements close to the centre are likely similar to CBR_L^0 as the radio propagation environment is not too different yet, more emphasis is put on the distance accuracy than the direction accuracy. With the growing distance of the one-hop measurement from the centre, emphasis shifts then to direction accuracy: The width of rings grows but the angular separation shrinks.

Starting on the innermost 'ring', which degrades to a circle, the sectors are assigned monotonously increasing numbers in the clockwise direction from 1 to 252. If no neighbour measurement is available within the maximum distance of 1000 m, the sector number 0 is used. The maximum of 1000 m has been chosen in analogy to the theoretical communication distance employed by GN. Three sector numbers in the upper range remain unused with this segmentation pattern. The employed segmentation is a compromise on sector size, accuracy in direction and accuracy in distance for the limited bits available. Other patterns may be equally suitable or even better in some constellations; however, this thesis aims at determining to which extent such directional information associated with sectors is beneficial at all.

Table 4.6: Properties of each ring employed to encode directional CBRs

Ring	Number	Distance range	Sectors
0	1–4	[0 m; 50 m)	4
1	5–16	[50 m; 150 m)	12
2	17–36	[150 m; 250 m)	20
3	37–68	[250 m; 400 m)	32
4	69–112	[400 m; 550 m)	44
5	113–172	[550 m; 750 m)	60
6	173–252	[750 m; 1000 m)	80

Field Generation by Sender

In contrast to the standardised information sharing, which always employs the maximum received CBR, a particular CBR received from a one-hop neighbour needs to be selected for inclusion in the outgoing SHB header. Many strategies to pick a one-hop neighbour are conceivable then.

A straightforward strategy picks a random candidate from the received $\mathrm{CBR}_{\mathrm{L}}^{0}$ fields within the last second and which are closer than 1000 m, *i.e.* representable by the outgoing $\mathrm{CBR}_{\mathrm{L}}^{1}$ sector field. Ease of implementation is one advantage of this strategy. Furthermore, it treats all suitable measurements fairly as each is equally likely to get picked by chance.

However, this selection procedure tends to include CBRs from areas with high vehicle density more often as more measurements are reported from there as well. An alternative could be to select CBRs from remote stations preferably. This approach then may become unfair, though, if the environment does not allow for similar transmission ranges in all directions, *e.g.* because of buildings obstructing radio propagation at one side.

To minimise the probability that CBRs from the same sector are repeatedly sent, the CBRs belonging to the same sector are aggregated. Taking the mean of all aggregated CBR samples per sector also prevents that contradictory measurements of the same sector are shared. Then, one of the sectors with aggregated samples is randomly selected. This strategy prevents that sectors with high vehicle density are used predominantly. Instead, each sector with available measurements is equally likely picked. Consequently, a larger area is covered by shared CBRs as the manifoldness of reported sectors grows.

This last-mentioned strategy is employed in the following. As a prerequisite, it needs access to all received DCC-MCO fields with directional CBR reports. These reports are stored in LocTEs as outlined next.

Field Processing by Receiver

On reception of an SHB packet, the included DCC-MCO field is stored in the LocTE for the respective sending station. If the DCC-MAP field is employed, however, it is not sufficient to hold only the last field received from

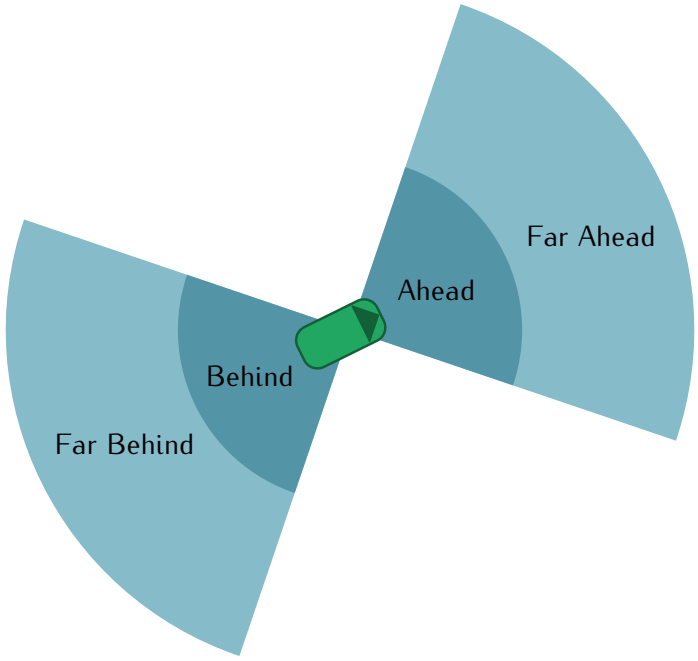


Figure 4.10: CBR queries exploiting localised information sharing

a station. Since stations are randomly selecting the one-hop CBR information for transmission, a number of received DCC-MAP fields should be buffered so a detailed map of CBR measurements can be reconstructed.

The reference implementation of DCC-MAP in Vanetza foresees a ring buffer in each LocTE to hold this received information. This ring buffer has a capacity of 16 elements, *i.e.* it can keep all information shared by a station for at least 1.6 s even if its CA service transmits at the maximum rate. Along with the raw DCC-MAP fields also the reception timestamps and the sender's positions are stored in the ring buffer.

Enhanced by those ring buffers, the LocT becomes a database of localised CBR measurements. Among others, it is possible to mimic the calculation of global CBR as employed by standard information sharing. For this purpose, the latest DCC-MAP from each LocTE is taken as input to determine the maximum CBR of all one-hop neighbours. Additionally, all ring buffers are scanned for DCC-MAP fields whose storage age is less than 1 s. The maximum CBR of two-hop neighbours is calculated from those fields' conveyed as one-hop CBRs. While the original information sharing aggregates one-hop CBRs at the sender side, sharing directional CBRs shifts the calculation of the maximum CBR to the receiver side. These queries neglect any directional information, but they allow to assess the impact of the less exactly encoded CBRs.

Taking advantage of localised CBR measurements, a station can now also determine the level of channel congestion ahead and behind it separately. Specifically, four CBR aggregations are considered further on: Ahead, Far Ahead, Behind, Far Behind. Each of them refers to collected CBRs in a

90° cone in front or at the rear of the ego vehicle, respectively. *Ahead* and *Behind* consider CBRs up to a distance of 500 m, *Far Ahead* and *Far Behind* up to 1000 m. The cones are aligned with the vehicle's heading as shown schematically in Fig. 4.10. The result of each query is the mean of the ten largest CBR measurements known in the respective cone. Of course, more sophisticated queries are conceivable as well. These four queries are sufficient, however, to evaluate if a significant CBR difference can be detected by vehicles at all.

For comparison with the original DCC sub-entity at network layer (*DCC_NET*) using DCC-MCO fields, the global CBR can be determined as well. Since DCC-MAP does not solely include the maximum one-hop and two-hop CBRs, stations have to scan in their LocTEs for the respective maximum value among received DCC-MAP fields. Based on the *itsGnCbrLifetime* constant equalling 1 s given by *TS 102 636-4-2* [195], DCC-MAP fields stored in the LocT longer than 1 s are skipped. The global CBR is then, as provided by Eq. (10) in Section 2.4.3, the maximum of the local, one-hop, and two-hop CBRs.

Simulation

The following proof-of-concept simulation employs the Griddy scenario in its 'plus islands' variant, *i.e.* with buildings obstructing LOSs on some parts of the map as shown in Fig. 3.5b. Along with the GEMV² radio propagation model, which considers obstacles such as buildings, the actual channel congestion varies then though all stations are generating CAMs at a fixed rate of 5 Hz.

Figure 4.11 depicts the local CBR measurements of each vehicle in the network between simulation time 91.9 s to 92 s. Since CBRs are measured at 100 ms intervals, every vehicle is represented by exactly one cross mark. These crosses highlight the position of the respective vehicle, and their colour indicates the measured CBR. For visual guidance, the CBR contours are plotted in the background. The colouring confirms that the local channel congestion between buildings is indeed considerably lower than in unobstructed areas. Furthermore, the vehicle density on road segments does not correlate with channel congestion. Areas with high vehicle density are highlighted by dark contours in Fig. 4.12 for comparison. For example, the intersection at position (400 m, 400 m) is crowded, but the channel congestion is well below the maximum congestion level observed across the whole map. Since all vehicles are generating CAMs at a constant rate, also while waiting at a stop line, this observation cannot be attributed to vehicle dynamics.

Figure 4.13 shows the global CBR determined on the basis of standardised DCC-MCO and the proposed DCC-MAP fields. When only 10 % of the vehicles are equipped with IVC, the global CBR is on average slightly above

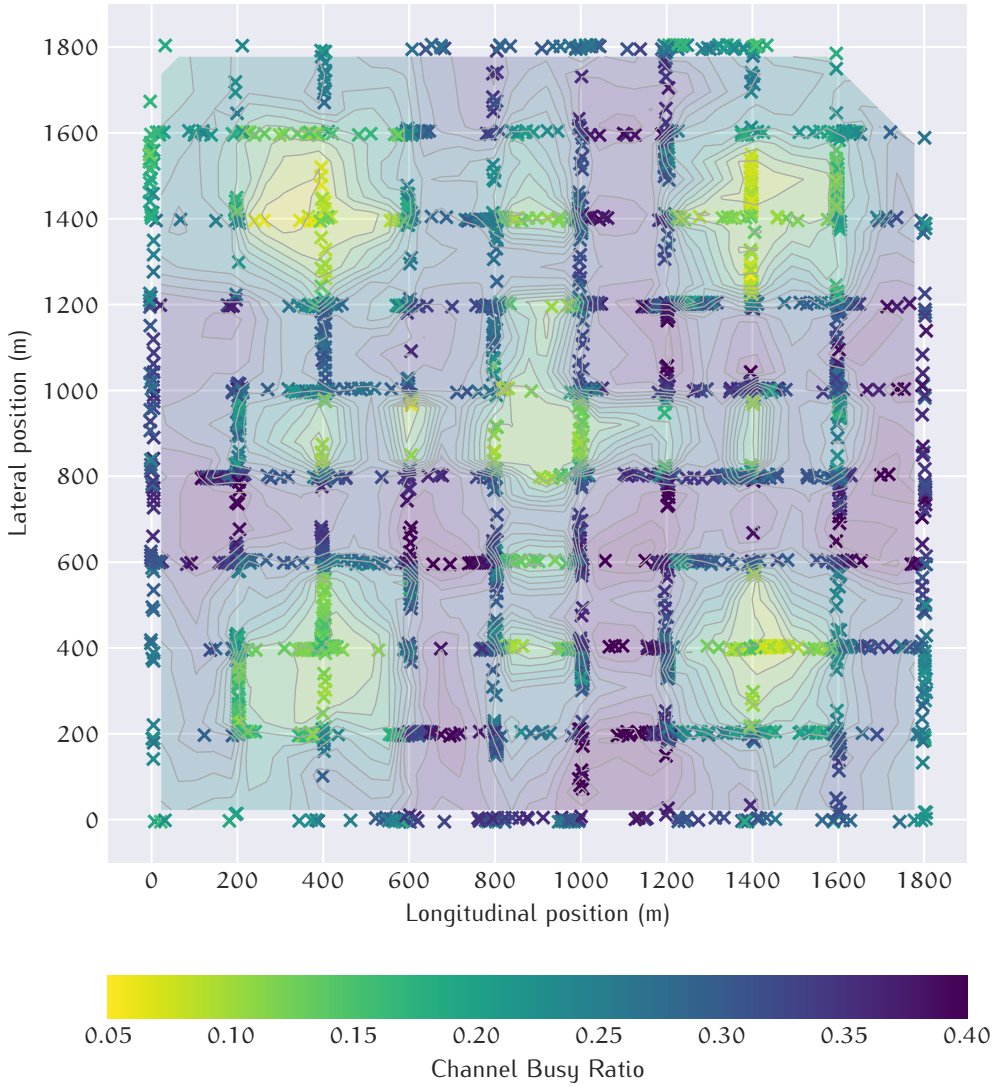


Figure 4.11: Local CBRs as measured by vehicles between 91.9 s to 92 s

13 %. With fully equipped vehicles, the global CBR is between 44 % and 45 %. The narrow error bands show a close agreement among all vehicles regarding the global CBR. Based on DCC-MAP, the global CBR is always lower than its DCC-MCO counterpart. Two reasons can be identified for this observation: First, low congestion levels below 30 % can only be encoded in 5 % steps with DCC-MAP. Hence, the encoded value for CBRs between 10 % and 15 % is identical. On the receiver side, the mean of the respective range is assumed for further processing, *i.e.* 12.5 % in this case. This procedure also explains the global CBR observed with the lower equip rate. Second, the shared one-hop CBRs are randomly selected when DCC-MAP is employed, as explained earlier. With an increasing pool of selectable one-hop CBRs, it becomes more likely that the maximum value is not selected in the past time window of 2 s. Hence, the global CBR determined via DCC-MAP remains 2 to 3 percentage points below DCC-MCO despite the increased resolution of encoded CBR above 30 %.

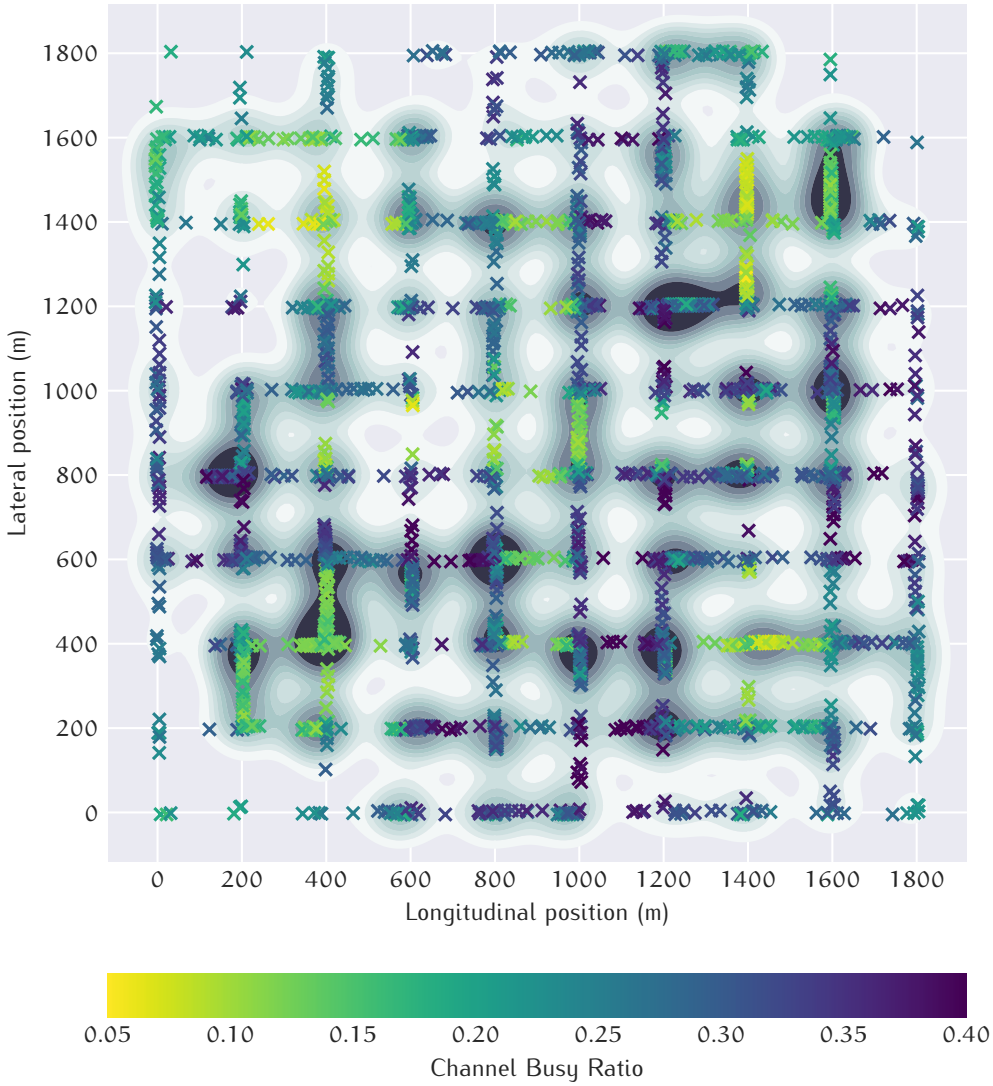
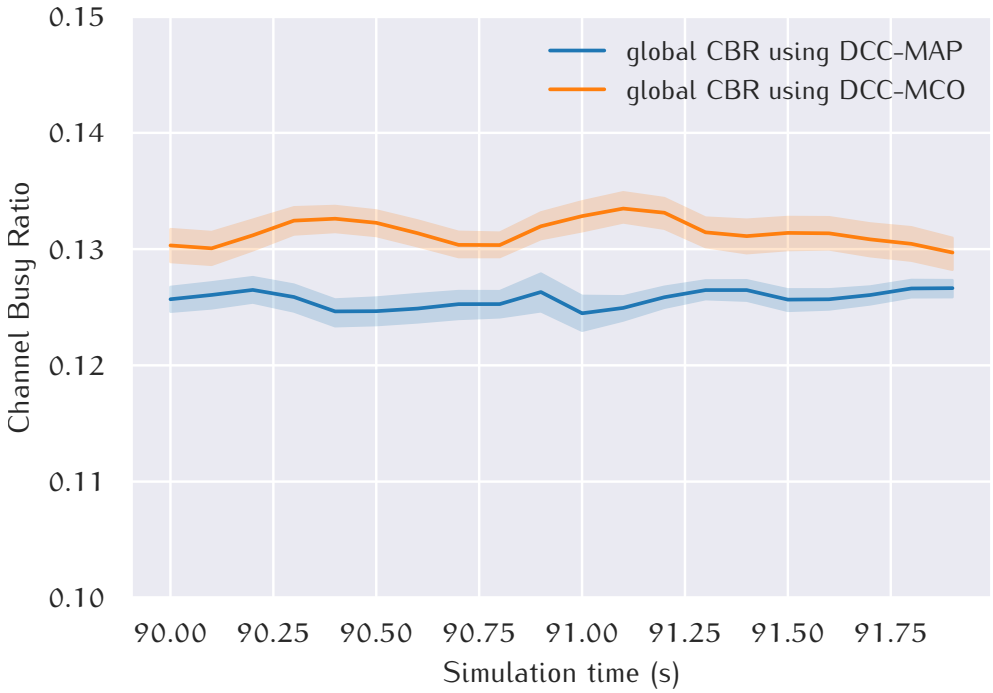
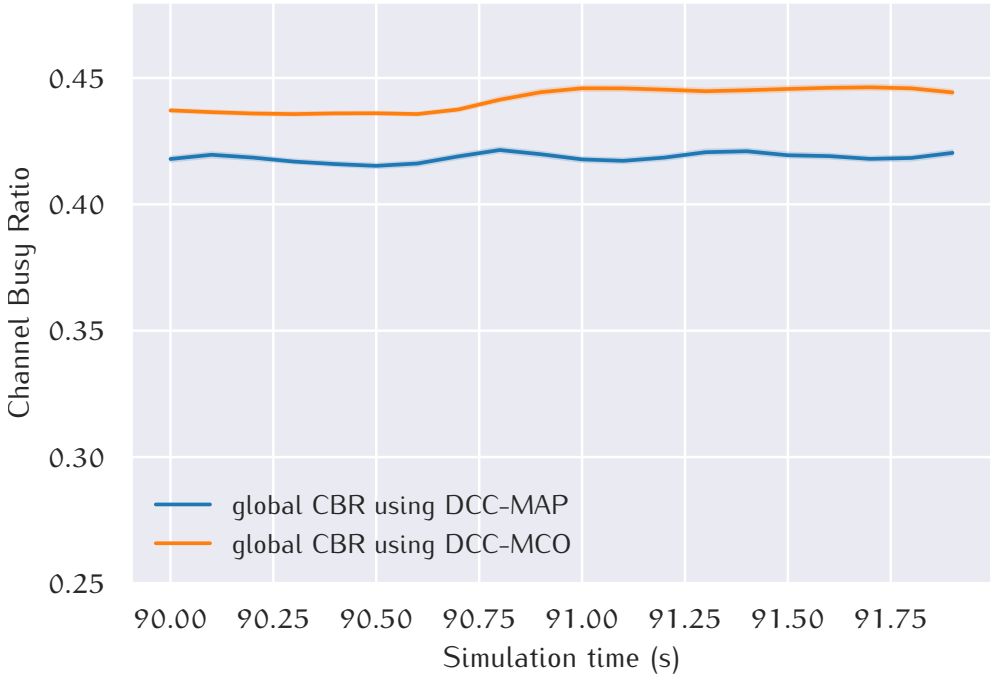


Figure 4.12: Vehicle density corresponding to local CBRs during 91.9 s to 92 s

The perspective of a particular *Ego* vehicle of its surrounding channel congestion via DCC-MAP is depicted in Fig. 4.14. This ego perspective corresponds to the global congestion situation shown in Fig. 4.11. In this dense traffic environment, this single inspected vehicle knows about 245 one-hop CBR measurements received within the last second. Furthermore, 1296 two-hop CBR measurements not older than 2 s are available in its LocT. The size of the two-hop CBR markers denotes the distance between the one-hop and the two-hop neighbour. These two-hop measurements are partially overlapping with regions where more accurate one-hop measurements are available as well. However, more distant regions are covered predominantly by those two-hop measurements. In the south-western map region, the ego vehicle also knows about the prevailing congestion level only by two-hop CBR reports; the same distance in west or south direction is covered by one-hop reports, though. Occasionally, one can also spot co-located two-hop CBR reports which indicate significantly different congestion levels. For example,



(a) Global CBR with 10 % equipped vehicles



(b) Global CBR with 100 % equipped vehicles

Figure 4.13: Global CBR based on DCC-MCO versus DCC-MAP

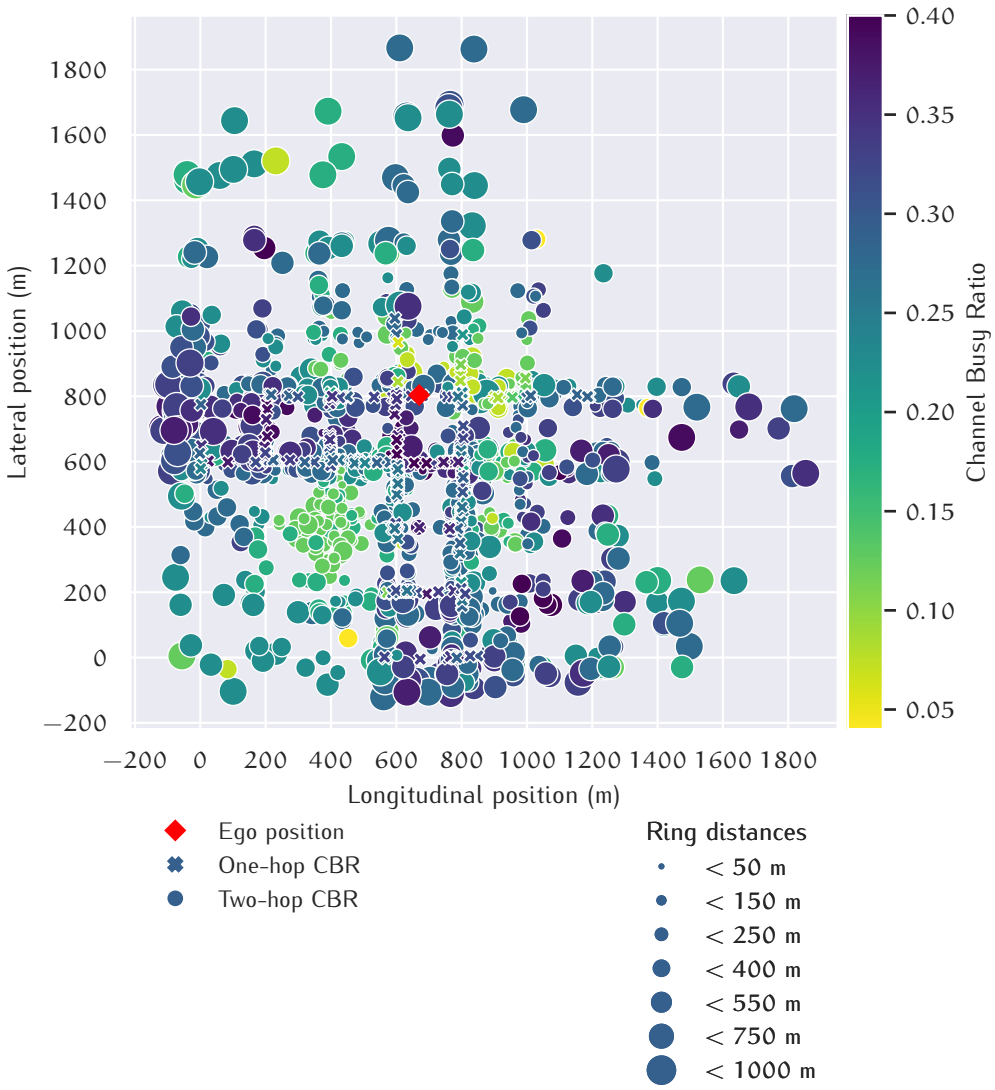


Figure 4.14: Shared CBR information from an ego perspective

roughly at position (1000 m, 1300 m) a yellow and blue marker are plotted which represent CBRs of about 5 % and 30 %, respectively. However, close to this position buildings form a kind of 'canyon' which shadows radio signals much stronger than the open space nearby. Furthermore, the position of a two-hop marker is always approximated based on the ring sectors shown in Fig. 4.9, *i.e.* such a reconstructed position can be easily off by tenths of meters. The combination of both can then lead to seemingly wrong, but yet reasonable CBR reports.

Concluding these preliminary findings, the global CBR calculated on the basis of DCC-MAP fields does not precisely resemble the outcome of DCC-MCO. However, the results are still close enough to consider DCC-MAP as an alternative to DCC-MCO further. Concerning a single vehicle's view towards localised channel congestion, the congestion levels restored from shared CBRs via DCC-MAP conform reasonably with the local measurements by each vehicle.

4.2.2 Flushing Store-Carry-Forward Buffers

According to the GN standardisation [193], the SCF packet buffer is flushed entirely when another station has come into transmission range, *i.e.* on any *GN-DATA.indication*. Section 4.1.4 presented SCF enhancements at the GN layer, which deal with the aspects if an SCF-enabled packet shall be buffered and when it is ready for continued dissemination. However, GN cannot determine at which rate it can safely flush its SCF buffer without flooding the network.

Problem Description and Basic Idea

By default, this buffer can hold up to 1 MB of data or over 400 packets assuming the maximum 802.11 frame length for each packet. Keeping in mind the mandatory pause of $T_{off} = 25\,ms$ after each transmission, the mere waiting time sums up to 10 s. One may argue that the buffer size is overly large and thus the specified head-drop policy of the underlying FIFO queue, *i.e.* dropping the oldest packets first, rarely comes into effect. In the interest to avoid buffer bloat, the queues at DCC_ACC shall be small since their purpose is to buffer arriving packets only briefly until the next transmission opportunity. *TS 102 687* [61] demands transmit queues to store not more than two packets per AC. Flushing a well-filled SCF buffer all at once thus just leads to immediate packet drop at the next lower layer.

Processing of SCF packets by GN can be enhanced by taking advantage that those packets are marked as delay-tolerant by upper layers in the first place. It is safe to assume that upper layers also tolerate a little bit more delay in favour of not discarding their packet. GN should not flush its SCF buffer all at once, but DCC should notify GN about available transmission opportunities to drain its SCF buffer step by step. Since SCF is designed to overcome fragmented networks when traffic is sparse, one can expect some free channel capacity for delay-tolerant packets as the volume of delay-sensitive packets is low. How DCC determines when to serve pending SCF packets is further studied in Section 4.2.5.

SCF Signalling between DCC and GN

On the one hand, GN stores SCF packets according to given policies, but it lacks profound knowledge when it is advisable to transmit a packet without unwanted side effects. On the other hand, DCC knows exactly about channel access constraints. Still, it is neither suitable to buffer many packets nor should it be tightly coupled with internal logics of upper layers. Some high-level signalling between GN and the DCC cross-layer can bridge this gap, fortunately. This signalling is enabled through a new interface called 'SCF valve', which grants DCC limited access to GN's SCF packet buffer. The

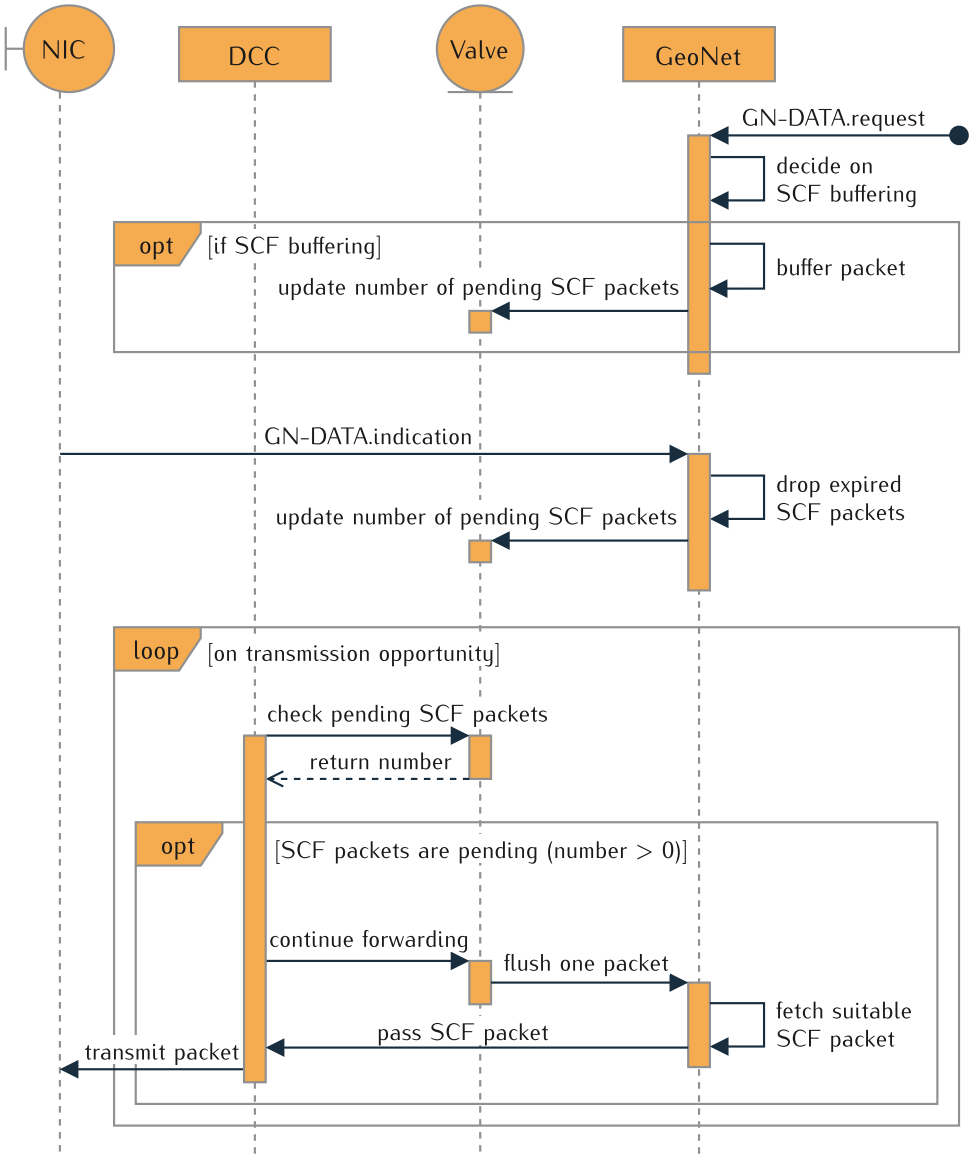


Figure 4.15: Interaction between DCC and GN through SCF valve

sequence diagram in Fig. 4.15 visualises the proposed modifications: The decision when a packet is going to be enqueued by SCF remains unmodified solely under GN's control. However, GN removes only expired packets from the SCF packet buffer when processing *GN-DATA.indication*. No SCF packets are flushed at this point anymore but only when DCC tells GN to flush.

DCC can retrieve the number of pending SCF packets through the valve interface. Consequently, DCC knows that it has to consider these delay-tolerant packets in its scheduling of channel access. When sufficient channel resources are available, DCC explicitly notifies GN that it can pass down one SCF packet at most now. GN then selects a suitable SCF packet according to its flush policy, if any is matching at all. This packet retrieval pattern avoids the necessity for another large buffer at DCC that can keep up with the size of the SCF buffer. Details of SCF buffering remain hidden from DCC, though.

Back in Section 4.1.4, the SCF flush policy got invoked upon packet receptions. Thus, the candidate MAC address required by the SCF flush policy is straightforward the source MAC address of the just received packet. With the introduction of the SCF valve, however, no single candidate station is at hand when DCC indicates a chance to disseminate an SCF packet. As a remedy, the MAC addresses of all stations marked as neighbours in the LocT will be considered as candidates.

4.2.3 Guaranteed Timing for Periodic Services

As it stands with *TS 102 687* [143], the gatekeeper is a rather simple mechanism located on top of the radio's MAC interface. It is meant to protect the channel from excessive packets when upper layers fail to maintain the channel usage limits. Hence, the gatekeeper may temporarily refuse to pass on scheduled packets for transmission. Depending on the implementation, the gatekeeper may drop packets or buffer them to throttle the transmission rate, *e.g.* using a leaky bucket algorithm.

Periodic services such as CA and CP, however, ask only for the current interval between consecutive packets of their traffic class to realise TRC [171, 175]. Applications sharing the same traffic class may thus race for channel access without feedback if their packet could be transmitted without delays. The proposed solution for this issue is an interface between applications and DCC, that allows applications to allocate channel resources before passing their payload down for transmission. If a transmission has been preallocated, DCC guarantees to handle the packet within the agreed constraints. For example, such a constraint can be that a message will not be enqueued by DCC if passed down within a given time window.

Characteristics of Periodic Services

The services are unable to allocate sufficient yet minimal resources themselves because they hardly know the final packet's length in advance. The structure of CAMs and CPMs has many optional fields or lists of variable sizes. Presence of these fields can depend on the time since last message generation, *e.g.* a CAM low-frequency container every 0.5 s, or the surroundings, *e.g.* the number of perceived objects to be included in a CPM. Furthermore, the overhead added by lower layers is not of fixed length. Retaking the CA service as an example, the full certificate will be added at least once every second by the security entity at the network layer. More frequent CAMs may only contain a short certificate digest instead. However, the decision to use a full certificate or just its digest is not taken by the service but lower layers.

Periodic services know their message rates, though, at least CA and CP services. These two services can thus announce to DCC that they want to transmit their messages every 0.1 s to 1 s. Furthermore, each service has a unique AID which is assigned according to *TS 102 965* [197]. Through the prior announcement, DCC at the link layer knows about running services and when they are willing to generate the next message earliest. DCC can then monitor channel usage of each service and assign them individual airtime budgets. In a particular implementation, the total airtime may be split evenly among services of the same DP, *i.e.* services emitting shorter packets are allowed to generate packets more often. Instead of services individually polling current TRC limits, DCC as cross-layer can support periodic services by calling them according to the Hollywood principle[1]. Since DCC_ACC tracks actual airtime usage of each service, also different characteristic message lengths among vehicle manufacturers are incorporated. Suppose a manufacturer's vehicle tends to produce longer CAMs than a competitor's CA service implementation. Then, its CA service is less often allowed to transmit under the same channel usage constraints than another vehicle's service generating shorter messages.

Allocation Interface

The interface from a service's perspective looks as shown in Fig. 4.16 and is described hereafter. During its initialisation phase, a service announces its AID and DP at the DCC cross-layer via *DCC_FAC.announce* for identification. The service also registers a callback at DCC_FAC, which will be invoked by DCC if a transmission opportunity becomes available for the particular service.

When the callback is invoked, DCC promises the service undelayed processing, *i.e.* its packet is not enqueued by DCC_ACC if it arrives at the link layer in the predefined time window. DCC_ACC can compare if the AID + DP pair of a transmission request matches the grant given by DCC_FAC to a periodic service. Furthermore, DCC_ACC is able to note the effective airtime consumed by each AID, which can then be considered for assigning future transmission opportunities. To enable matching the AID + DP pair, the original interface between network and link layer specified in *TS 102 723-10* [73] requires a minor extension: Not only the DP but also the AID needs to get passed. Services can further employ the existing, unmodified interface to pass down their payloads: A service invokes BTP via `BTP-Data.request`, which is the standardised BTP data service primitive for transmissions [170]. Noteworthy, the BTP service primitive has no notion of an AID either but of a destination port. Since services have to use well-known port numbers, a one-on-one relation exists between port numbers and AIDs. This mapping is

1 also formally known as inversion of control

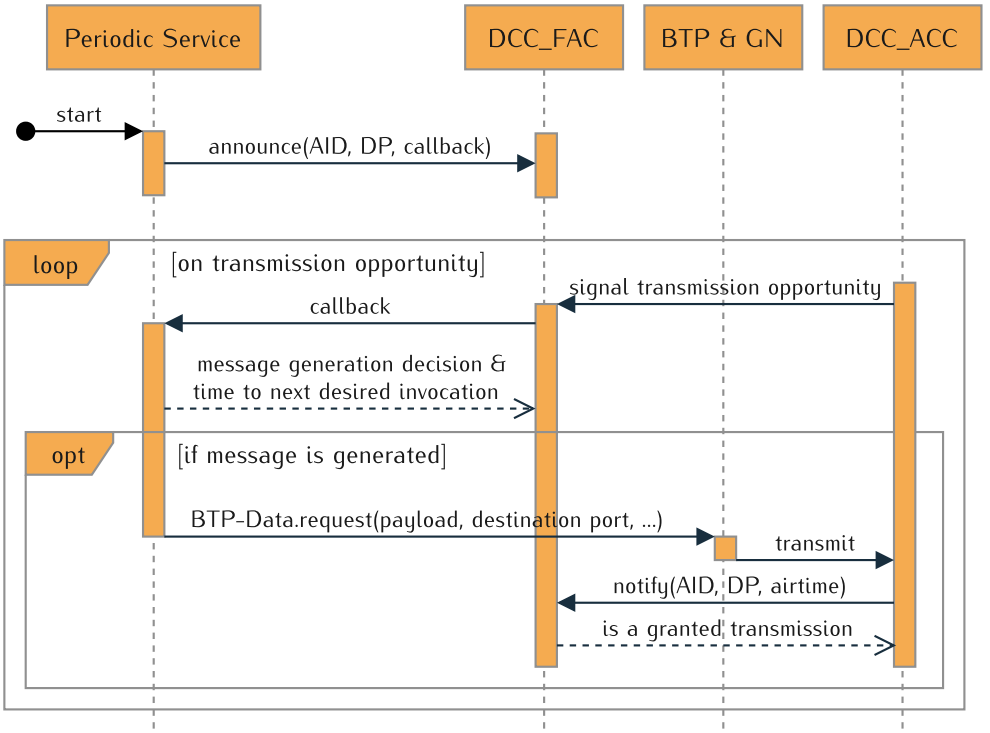


Figure 4.16: Enhanced DCC handling for periodic services

already employed when security profiles depending on packet's AID are applied at the network layer. Hence, the allocation interface can build upon this existing mechanism without the need to extend the BTP service primitives.

Invocation of the callback does not force the service to generate a packet, though, it is instead an offer by DCC. Hence, the return value of the callback indicates if the service is going to use an offered transmission opportunity and when it wants to be considered for another transmission next time. For example, if vehicle dynamics do not demand a CAM at the moment, the service can pass on the offered channel access to another service. Thus, a service can cooperatively allow sibling services to use its resources for now. Waiving offers shall not penalise a service, however, for example by preferring collaborative services at the next transmission opportunity.

Whereas CA and CP services as standardised poll the currently allowed message generation rate by DCC periodically, the proposed allocation interface allows notifying these services when message generation is in fact possible. This inversion of control shifts the responsibility to manage channel access from individual services to the scheduler entity of DCC, which is detailed in the follow-up Section 4.2.5.

4.2.4 Forwarding Traffic Awareness

Successful forwarding in VANETs depends on the collaboration of its network nodes. However, stations have no direct benefit when they forward a packet as they have already received a particular packet's information. If channel resources become scarce in congested scenarios, they can decide to omit forwarding in favour of their own packets. For example, the BSP of C2C-CC explicitly downgrades forwarding traffic to the lowest priority DP3 irrespective of its original priority by the source station [163]. Depending on the employed queue policy, DP3 traffic may starve as CAMs with DP2 are continuously generated.

Even if DP3 packets do not starve entirely in their DCC_ACC queue, Kühlmorgen et al. [200] point out that the delays induced by this enqueuing break the overhearing of CBF. In essence, when DCC enqueues a CBF packet its transmission cannot be aborted by GN anymore even if it recognises forwarding transmissions of the same packet. Kühlmorgen et al. introduce a tight coupling named 'RORA' between DCC and GN to mitigate this problem: DCC's gatekeeper has to check for meanwhile received GBC duplicates at GN when dequeuing a to-be-forwarded packet. ETSI adopts their mechanism in an informative annexe of *TS 102 636-4-2* [195], where it is limited to plain CBF though. Hence, bounded redundancy, as explained in Section 4.1.1, is not supported right away.

It would be feasible to generalise 'RORA' to support also advanced CBF features such as bounded redundancy. The approach in this thesis, however, is to prevent the described problem caused by the DCC gatekeeper in the first place. Motivated by prior work on bufferbloat (see Section 2.3.3), it seems worth striving to avoid enqueuing of forwarding packets by DCC_ACC whenever possible.

Forwarding Budget

Avoiding queues at DCC for forwarding packets, which may occur at any time is only possible by violating the $T_{off} = 25\,ms$ gap between transmissions mandated by *EN 302 571* [128] deliberately. If this decision, along with an adapted set of rules regulating transmission, proves to improve QoS in ITS-G5 network, it may be advisable to lift this restriction imposed by standardisation.

As it stands now, stations have almost no incentive to forward packets as these packets compete with own packets for channel access. Hence, accounting of resource usage for packets generated by a station and those forwarded as collaborative effort should be split up. This idea leads to the concept of a dedicated forwarding budget, which is shared among stations for packet forwarding transmissions.

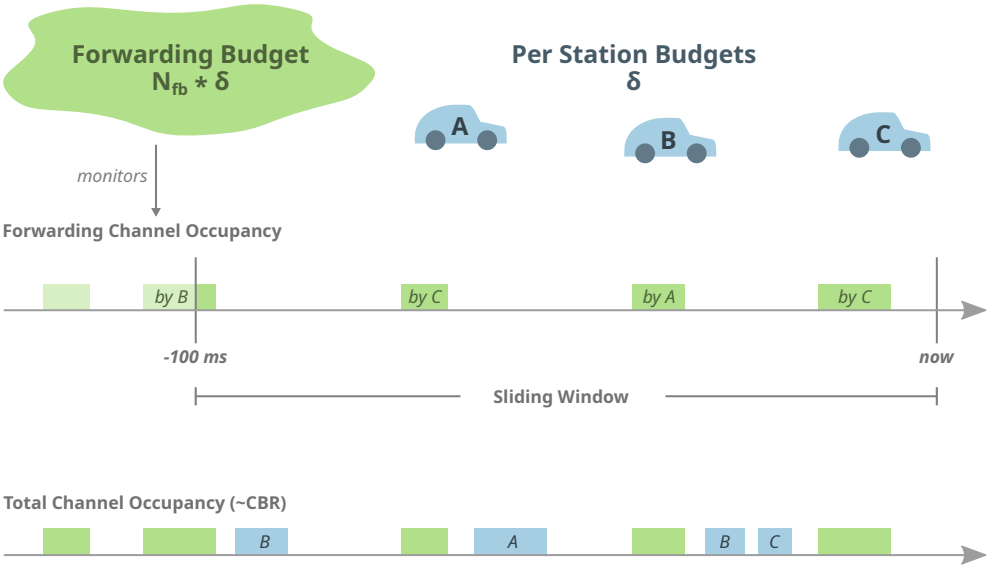


Figure 4.17: Forwarding budget and monitoring of channel occupancy

GN itself limits the packet data rates as part of its forwarding operations. The default limit in standardisation is set to 100 kB/s per station [193]. Stations track the data rates of each other in their LocTs and omit to forward if the defined limit is exceeded. Limit checks are applied on the reception of GBC packets for the current sender and the original source station. The sender is identified by the source MAC address, the original source by the source position vector in GBC's extended header. At the initial dissemination step, sender and source station are identical obviously. Since GN prevents excessive packet forwarding caused by particular stations, the risk is low that the forwarding budget is consumed only for a single source. Hence, DCC shall not reject any forwarding packet deemed worth forwarding by GN.

DCC is responsible for determining the currently applicable channel occupancy limit for packet forwarding. For this purpose, adaptive DCC presented in Section 2.4.2 is customised. Originally, adaptive DCC aims to converge to equal occupancy limits δ for all nearby stations. The forwarding budget can then be seen as some virtual stations, which do not emit CAMs and alike but only forward packets. Experiments regarding a suitable number of virtual stations N_{fb} are conducted in Chapter 5. Since the underlying LIMERIC algorithm already adjusts each station's δ, the forwarding budget's occupancy limit $N_{fb} * \delta$ is also dynamically adapted to current congestion levels though N_{fb} is fixed. Depending on the selected N_{fb}, the operative forwarding budget ranges from at least $N_{fb} * \delta_{min}$ to $N_{fb} * \delta_{max}$ at maximum. According to standardisation, the two LIMERIC parameters are set to $\delta_{min} = 0.0006$ and $\delta_{max} = 0.03$ [143].

Determining forwarding budget's duty cycle limit is just one side of the coin, though, as schematically shown in Fig. 4.17. GN has to fulfil two jobs when the proposed forwarding budget is employed. First, it has to employ

this budget for own forwarding transmissions instead of the traditional DCC budget. Second, GN has to track the amount of channel usage by other stations consumed for forwarding. As the forwarding budget is a shared, cumulated budget among all stations, stations have to enter own as well as foreign forwarding transmissions on this budget. Whether a received packet has been forwarded can be determined as follows: First, the packet's type must indicate a multi-hop packet like GBC. If the sender's MAC address then does not match the address in the source position vector of GBC's extended header, the packet has been sent by a forwarding station. GN monitors channel occupancy by these packets in a sliding window for the last 100 ms.

Elastic Budget Calculation

The budget factor N_{fb} is fixed to guarantee a minimum budget available even in congested networks. However, stations may be too few in sparse networks to reach the CBR_{target} even if they were all occupying the channel with their maximum duty cycle δ_{max}. In these uncongested networks this CBR gap, which cannot be filled by source packets at all, can be safely designated as additional forwarding budget.

Relying on a well-maintained LocT, *i.e.* with the enhancements of Section 4.1.2 in charge, a station can determine the current number of neighbours n_{LocTE}. Since LocTEs flagged as neighbours will only expiry after a few seconds, the reported neighbour count can be considered an upper bound, which is usually slightly above the actual number of neighbours. The forwarding budget can be safely expanded according to Eq. (19) without the risk that the forwarding budget interferes with channel resources allocated for original source packets.

$$\text{Forwarding budget} = \max\{N_{fb} * \delta,\ CBR_{target} - n_{LocTE} * \delta_{max}\} \tag{19}$$

Equation (19) is a conservative budget expansion because it assumes the maximum duty cycle δ_{max} for every neighbour. A further, less conservative budget expansion can make use of channel capacity not used by stations otherwise. This expansion mechanism is called 'elastic' because it borrows unused channel capacity as long as it remains unused. However, this mechanism also shrinks its budget proactively as general channel usage rises. For this purpose, a long-term CBR named CBR_{lt} is determined, which represents the maximum CBR observed during the last second. As long as CBR_{lt} is below a predefined elastic budget threshold EB_{th}, the forwarding budget is allowed to grow by $EB_{th} - CBR_{lt}$. This thesis employs $EB_{th} = CBR_{target} - N_{fb} * \delta_{max}$ in the following. Consequently, if the long-term CBR is low enough that even a fully exploited forwarding budget cannot reach CBR_{target}, then the budget

is elastically expanded by the respective CBR gap. In dense networks, where n_{LocTE} and CBR_{lt} are high, the forwarding budget stays at original $N_{fb} * \delta$ as a foundation.

Transmission Parameters

To make most out of the forwarding budget also in congested scenarios, GN can adjust the transmission parameters when forwarding. Since the forwarding budget is limited in terms of its maximum channel occupancy, the available airtime is restricted. The required airtime per forwarding can be adjusted by selecting a higher data rate from the available MCSs. In the context of packet forwarding, the following TDC strategy is proposed: With an increasing usage level of the forwarding budget and decreasing importance of a forwarding packet, higher data rates will be picked progressively. Higher data rates shorten the channel occupancy of a packet in terms of airtime, but the transmission's robustness declines as well because a better SNIRs is required to decode the packet successfully. Hence, it is preferable to select robust MCSs as long as plenty of the budget is left. Yet, depletion of the budget has to be prevented as no further forwarding transmissions are allowed then.

The data rate selection algorithm is detailed in Eq. (20). First of all, the $importance$ of a particular forwarding is evaluated. This importance diminishes as the CBF redundancy grows. For plain CBF, where a packet is forwarded at most once by a station ($c_{max} := 1$), the CBF counter for this packet is always $c_p = 1$. For CBF with bounded redundancy (see also Section 4.1.1), $c_{max} > 1$ and $1 < c_p <= c_{max}$ applies. The $importance$ is then combined with usage level of the forwarding budget, denoted as fb_{usage}, to derive the rate selection variable $select$. In the case of plain CBF, $select$ simply equals fb_{usage}. For redundant forwarding transmissions, where $importance < 1$, $select$'s magnitude is amplified. Finally, a specific $rate$ is looked up for $select$, with higher data rates for larger $select$ values.

$$
\begin{aligned}
importance &= 1.0 - \frac{c_p}{c_{max}} \\
select &= fb_{usage} * (2 - importance) \\
rate &= \begin{cases}
6\,\text{Mbit/s} & \text{if } select < 0.34 \\
9\,\text{Mbit/s} & \text{if } select < 0.56 \\
12\,\text{Mbit/s} & \text{if } select < 0.73 \\
18\,\text{Mbit/s} & \text{if } select < 0.84 \\
24\,\text{Mbit/s} & \text{if } select < 0.93 \\
27\,\text{Mbit/s} & \text{otherwise}
\end{cases}
\end{aligned}
\tag{20}
$$

Rates below the ITS-G5 default rate of 6 Mbit/s are never picked because they inevitably result in longer airtimes than without adaptation. Please note that the *select* steps to cut down the airtime by half are constantly decreasing as the budget consumption grows: While an increase by $0.73 - 0.34 = 0.39$ is necessary to switch from 6 Mbit/s to 12 Mbit/s, doubling the rate again to 24 Mbit/s is only a further increase by $0.93 - 0.73 = 0.2$. Other values could be selected alike, as long as the step sizes of *select* are progressively reduced. As the channel occupancy by forwarding fills up, higher data rates need to be selected, so stations make room eagerly for eventually more forwarding transmissions.

As mentioned before, fb_{usage} is determined by GN by monitoring own and foreign forwarding transmissions. However, the value of fb_{usage} is not the absolute channel occupancy, *i.e.* the total airtime of forwarding transmissions in the last 100 ms, but the ratio of this occupancy to the current budget. For example, assume a budget of $N_{fb} * \delta = 0.2$ and ten forwarding transmissions each enduring $T_{on} = 1$ ms. The absolute channel occupancy by forwarding is $^{10\,ms}/_{100\,ms} = 0.1$, but fb_{usage} is $^{0.1}/_{0.2} = 0.5$.

If the remaining forwarding budget still becomes insufficient, the behaviour depends on the forwarding algorithm. Classic area- and non-area forwarding such as CBF will simply omit to forward packets if no resources are left at the respective time point. Glow Forwarding, which is by design delay tolerant, can defer its forwarding operation in hopes that forwarding budget becomes available again soon. Such a deferral can be realised by rescheduling the Glow Forwarding timer. Referring to Algorithm 4, such a timer is usually reset in the range 0.75 to 1.25 of the packet's Glow Rate. If no forwarding has been possible due to insufficient forwarding budget, the timer is reset in the range 0.25 to 1.25 instead. Thus, a deferred Glow Forwarding transmission is retried quicker in such cases.

Summary

Forwarding traffic can be transmitted without delay if the $T_{off} >= 25$ ms transmission intervals are abolished in this particular case. Two mechanisms are in place, however, to avoid excessive channel usage: GN as standardised already demands to drop packets if packet data rate limits are exceeded. Furthermore, an upper bound for the duty cycle of all forwarding transmissions is introduced via the forwarding budget. This budget is commonly at disposal for any forwarding traffic and not reserved for a particular station. Stations acting collaboratively in forwarding thus have not to fear that their own packets are penalised when they opt to transmit a forwarding packet.

The available budget is determined dynamically according to a customised, adaptive DCC approach. Consequently, the available airtime for forwarding is as fairly distributed as the duty cycles for individual stations. If many for-

warding transmissions are necessary, and thus the budget consumption rises, switching to higher data rates aims to create more transmission opportunities for packet forwarding.

Chapter 5 evaluates all these measures in detail. In particular, the impact of the forwarding budget on other traffic needs to be studied, *e.g.* if CAM rates are reduced. Furthermore, the minimum guaranteed forwarding budget in congested networks can be tuned by N_{fb}. This parameter denotes how important packet forwarding is considered in a VANET in relation to single-hop communications.

4.2.5 Low-Latency Access Control

With the preparatory steps from Sections 4.2.2 to 4.2.4 in place, the remaining missing piece is their fusion to an upgraded DCC entity. In particular, DCC needs to schedule when to assign channel access to which sub-entity for serving the transmission demands of a station. Since a separate forwarding budget has been introduced in Section 4.2.4, such a scheduler is relieved from dealing with forwarding traffic and can focus on source traffic. All packets originating at a station are described as source traffic afterwards. Constraints for source traffic are a maximum duty cycle of 3 % and T_{off} gaps of at least 25 ms between consecutive transmissions [128]. Whenever these constraints are not violated, a station has a transmission opportunity from DCC's perspective. It is then DCC_ACC's decision if and for which purpose such an opportunity is used, though.

Objectives

The designs of current gatekeepers, which control the channel access ultimately at DCC_ACC, are comparatively simple as found in standardisation [143]. Either a fixed rate is enforced depending on the state of reactive DCC or the gate is reopened in intervals, which spread transmissions evenly to meet the occupancy limit determined by adaptive DCC. These mechanisms have already been discussed in Section 2.1.4.

Packets generated in close coordination between DCC and the respective service shall benefit from preferential treatment by DCC_ACC. In the context of this thesis, periodic services are encouraged to employ the DCC_FAC interface introduced in Section 4.2.3 in return of *minimised latency until channel access*. TRC is not enforced at DCC_ACC for those packets because the rate is already controlled per cross-layer interaction.

The arrival of source packets at DCC_ACC while no transmission opportunity exists require to buffer the packet temporarily or drop it immediately. Since transmission opportunities occur less frequently due to T_{off} than in conventional WLANs, buffering is recommended so event-triggered mes-

sages are not likely to get dropped, for instance. Forcing services such as DEN to align their packet generation with DCC_ACC would unnecessarily complicate their implementation otherwise. Sojourn time of packets in DCC queues is expected to be short, though. Explicitly delay-tolerant packets are managed by upper layers, *e.g.* by Glow Forwarding or SCF, where appropriately sized buffers are maintained for this purpose. Hence, DCC can *keep buffering capacities small* to avoid unfavourable effects associated with bufferbloat as discussed in Section 2.3.3.

DCC shall further support the operation of 'legacy services' not using any DCC_FAC mechanism. Minimising the channel access delay means implicitly that channel usage limits such as T_{off} are applied as late as possible. TRC, as enforced by DCC_ACC, has thus to be decoupled from TRC as advised to 'legacy services'. For example, the standard CA service queries for the allowed interval T_GenCamDcc of its DP. This advice by TRC should preserve the state-of-the-art semantics aiming at an even, average rate.

As announced in prior sections on timing guarantees for periodic services and stepwise flushing of SCF buffers, these features depend on the collaboration of DCC_ACC. These novel mechanisms need to be integrated in a manner that enables their operation but also serves DCC queues. Thus, DCC has to balance the assignment of source transmission opportunities among its queues, announced periodic services and pending SCF packets. Furthermore, DCC_ACC shall not interfere with transmissions covered by the forwarding managed, which is already managed by DCC.

In the following, this enhanced DCC_ACC entity for *F*orwarding *A*ware *L*ow *L*atency congestion *Co*ntrol is nicknamed 'Fallco'. Fallco's summarised objectives are:

- minimise latency by using transmission opportunities as quickly as possible
- avoid excessive buffering
- integrate improved mechanisms such as SCF forwarding valves and timing guarantees for periodic services
- backwards compatibility for legacy service designs, *i.e.* those not using any DCC_FAC mechanism or plain TRC

Inherited Concepts

Most details about the queues at DCC_ACC have vanished from [61] to [143]. In alignment with *TS 103 175* [107], for each AC, a corresponding DCC queue is put in front. Since current DCC_ACC specification [143] defines no queue length, the proposed length of two packet slots per queue from its predecessor is taken [61]. These queues are served strictly by priority

according to *TR 101 612* [96], *i.e.* DP1 packets are only dequeued if no DP0 packets are enqueued and so on. As mentioned in Section 2.1.4, Weighted Fair Queuing has no striking benefit over this Simple Priority Queuing policy [200].

Since LIMERIC, the adaptive DCC approach, is a prerequisite of the forwarding budget, it is convenient to govern a stations' budget for source traffic by this algorithm as well. Inspired by C2C-CC's BSP [163], an auxiliary budget for emergency messages is installed. This auxiliary budget allows on-top bursts of DP0 messages occurring within one second. These bursts are allowed to reoccur once every 10 s, *i.e.* are only suitable to disseminate messages triggered by a safety-critical situation but not for regular communication. Because LIMERIC aims to operate at the defined CBR target level whenever possible, it makes no sense to allow this auxiliary budget only during 'relaxed' channel congestion. Channel congestion slightly above the CBR target is accepted in the following if caused by emergency messages.

Though packet rates as determined by the gate opening algorithm Eq. (2) are not enforced by DCC_ACC anymore, services can still query them as a guideline. LIMERIC determines the available airtime for source traffic of DP1, DP2, and DP3, *i.e.* stations can be expected to have very similar budgets presuming δ converges. If some of these stations operate more services than others, they will have to determine how to spread capacity among them internally. Possible solutions for this purpose have been presented in Section 2.4.4. However, it can be still considered fair that each station's budget does not grow with the number of services.

As outlined in Section 2.3.5, packet traffic is sometimes pooled as flows in non-ITS networks. For example, a flow can be a particular end-to-end session between two hosts or even more specific two applications identified by port numbers on these hosts. Packet attributes such as AID and TC are candidates to adopt this flow concept for an ITS-G5 VANET. For example, a packet with AID 37 is known to be a DENM. Emergency and non-emergency DENMs can be distinguished by looking at the respective DP encoded in their TC. However, the particular DEN use case as determined by the cause code encoded in the DENM cannot be distinguished. Instead of a 'deep packet inspection' by decoding the DENM in the DCC cross-layer, it may be favourable to use currently unused TC-IDs if a more fine-grained differentiation becomes necessary. The TC-IDs 0 to 3, which correspond with DP0 to DP3, could be considered as 'primary priority' linked to the ACs. The currently unused 4 most-significant bits of the 6-bit-wide TC-ID field could be interpreted as 'secondary priority'. So far, no necessity for this level of differentiation has been found. In this thesis, shaping traffic by flows is omitted in favour of scheduling message generation by periodic services.

Scheduling

Three aspects are vital in Fallco's control of channel access:

1. How does Fallco deal with arriving transmission requests?
2. How to determine the next transmission opportunity?
3. For which specific purpose is an opportunity used?

ON TRANSMISSION REQUEST Upon the arrival of a packet requested to transmit by an upper layer, namely GN in ITS-G5, Fallco first checks if the packet matches a grant given to particular periodic service. Such a grant allows to conditionally bypass the DCC queues as guaranteed by the mechanisms introduced in Section 4.2.3. Two preconditions need to be met: First, the packet's AID and DP match those of the grant. Second, the packet has arrived within 25 ms since the grant has been awarded to the originating service. Only then DCC fulfils its promise and passes this packet immediately to the NIC.

Otherwise, DCC_ACC has to enqueue the packet unless an immediate transmission is possible nevertheless. Immediate transmissions are allowed for request without prior grant if all of the following conditions are fulfilled:

- station's channel occupancy (duty cycle) is below the limit δ
- no packet of equal or higher DP is already enqueued
- except for DP0 emergency packets, no bypass packet is pending
- packet intervals as advised by TRC are met

DETERMINING TRANSMISSION OPPORTUNITIES With the aim to enable channel access with minimum delay, the earliest time point for the next transmission opportunity has to be determined. Because of the $T_{off} >= 25\,ms$ constraint, DCC_ACC stores the time point at which it has passed a packet to the NIC last. Furthermore, it projects the time point when the station's duty cycle falls below the current channel occupancy limit. This projection determines how far the sliding window of 200 ms width needs to be shifted in the future, so enough prior transmissions drop out until the duty cycle undercuts its limit. The size of this sliding window corresponds to the update interval of LIMERIC. The maximum of the last transmission plus 25 ms and this projected time point is considered as next transmission opportunity.

If both time points lie in the past, *i.e.* the last transmission dates back longer than 25 ms and the duty cycle is currently below its limit, the algorithm falls back on the announced services and TRC. Thus, either enqueued packets will be served according to the rate advised by conventional TRC, or the next best

service's callback can be invoked, whatever comes first. As a last resort, if all queues are empty and no services have been announced, no transmission opportunity needs to be scheduled for the moment. Transmission opportunities are then reevaluated when DCC_ACC receives a request or a service is announced.

This reevaluation also takes places when LIMERIC recalculates the permitted duty cycle, which occurs every 200 ms. Enqueued packets may then gain earlier channel access according to the gate-opening update as per Eq. (3). Vice versa, their access time point may also be deferred further if the duty cycle limit is cut down. However, this affects only traffic governed by TRC but not any packets admitted by DCC_FAC.

ASSIGNMENT OF CHANNEL ACCESS The activity diagram in Fig. 4.18 outlines the decisions and actions taken by Fallco when a transmission opportunity is triggered at one of the previously described time points. Before any channel access is considered at all, the station's current duty cycle is checked against the duty cycle limit in force. Only if the limit is not exceeded, transmissions are considered at all. As long as the station's channel occupancy is too high, only the next triggering time is scheduled at the end. Each station stores the airtime of all its source transmissions for 200 ms so it can determine its duty cycle within this time window.

Otherwise, when channel capacity is still available, the state of DCC decides on further steps. Non-empty queues are then served in the order of their priority and at the rate given by TRC. Here, the TRC is given by the burst budget for emergency messages and the gate-opening intervals for other traffic. Should transmission opportunities occur more frequently than TRC allows for a particular DP, then packets in those queues are deemed as not ready for transmission. Absence of enqueued packets ready for transmission allows flushing an SCF packet if any are pending.

To guarantee the undelayed handling of messages admitted by DCC_FAC, DCC_ACC can temporarily refrain to dequeue packets or to flush an SCF packet. A message generated at the command of DCC_FAC can bypass the queues then as channel access for other purposes is locked. If the bypass path shall be enabled depends on the announced periodic services. DCC_ACC can only bypass one at a time, so if the bypass guarantee has been granted, the bypass path remains active for 25 ms at most or until the packet with matching AID and DP arrives. Though no new bypasses will be granted if the channel occupancy limit is hit, previously granted bypasses are still transmitted if they arrive within the given time frame at DCC_ACC. To avoid the starvation of any queues, services of a particular DP are only offered a bypass if the competing queue is empty or the average bypass rate per service

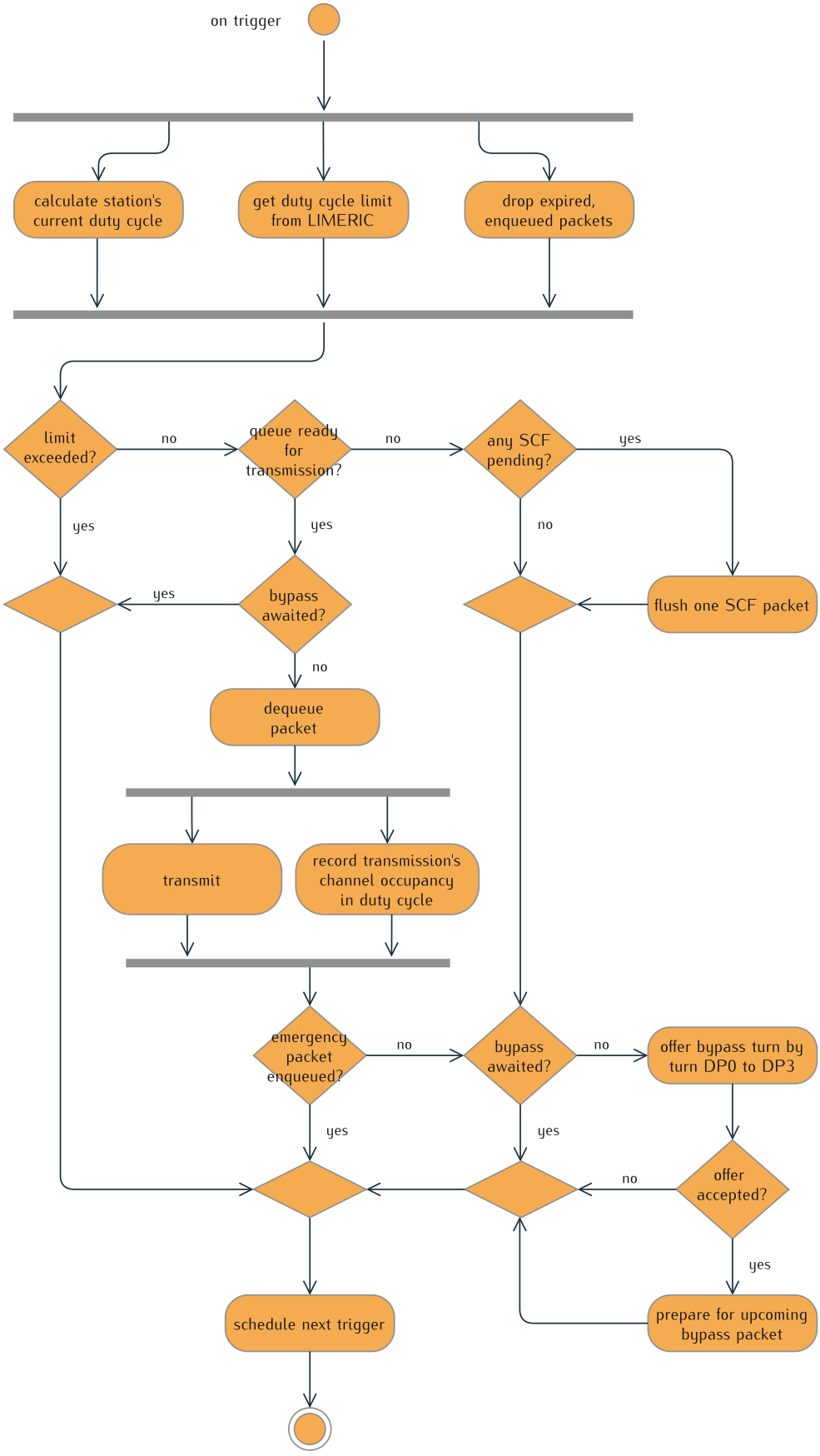


Figure 4.18: Actions taken by *Fallco* DCC on transmission opportunity trigger

is less than the packet arrival rate of this queue. Please note that emergency messages are still treated with preference by DCC_ACC and no bypass will be granted as long as any DP0 packet is pending.

Periodic services are offered transmission opportunities turn by turn with decreasing priority of their announced DP. Hence, a DP2 service receives an offer only if no DP0 or DP1 service is announced or willing to take the offer at the moment. Furthermore, all services are skipped, whose next desired invocation is after the current transmission opportunity. The next desired invocation time point is indicated by the return value of a service's callback function. Finally, the set of services willing to get a transmission opportunity right now is sorted by their channel occupancies in ascending order. Whenever a packet enters the bypass path at DCC_ACC, its airtime is booked on the respective service's channel occupancy. This bookkeeping is similar to the station's entire channel occupancy, however, with a sliding window of 1 s. Consequently, services generating shorter messages or less frequently are offered opportunities first. If they pass on the opportunity – indicated by their callback function – the next service with the next higher channel occupancy gets a chance.

Conclusion

The combination of the described processing of source traffic with the forwarding budget enables Fallco to pass on dedicated traffic with minimal delay to the underlying NIC. On the one hand, the information age for periodic services is then kept as small as possible. On the other hand, DCC avoids to handicap forwarding transmission as effectively as possible.

Furthermore, the risk that SCF transmissions suddenly overflow DCC queues is reduced by flushing their buffer located at GN in a controlled manner via the SCF valve. Messages triggered by events or by legacy services, which cannot or do not want to collaborate with DCC, are still supported in this DCC setup. Short queues still exist for these messages and are served regularly following the TRC scheme known from standardisation. Figure 4.19 highlights the miscellaneous components which make up the Fallco DCC cross-layer.

Despite all the efforts culminating in the presented, upgraded DCC, in the end, the total airtime in a single channel cannot be expanded but only distributed as smartly as possible. High-demand scenarios in Chapter 5 allow studying whether a more reasonable compromise is achieved with Fallco compared to the state-of-the-art DCC approaches.

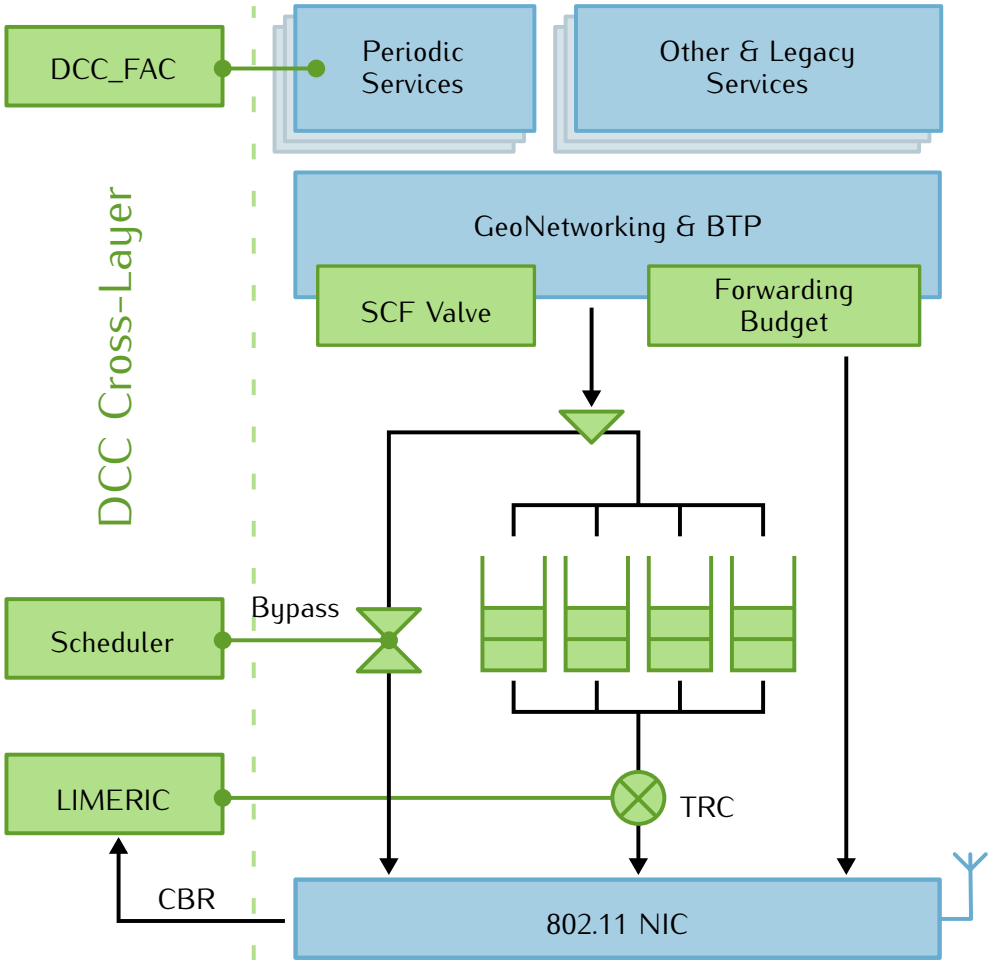


Figure 4.19: Components of the Fallco DCC cross-layer

4.3 SUMMARY

It is not a particular layer of the ITS-G5 design that solely constraints a VANET's message dissemination performance. Instead, efficient and reliable message dissemination strongly relies on the collaboration of network participants: They collaborate to keep the joint network load as low as possible and help to spread their messages mutually by forwarding. These two aspects of collaboration among ITS-G5 stations are found in the GN and the DCC layers, respectively.

The enhancements related to GN can be headlined by avoiding channel access whenever possible: CBF may waste channel capacity by quickly repeating redundant packets. Glow Forwarding, however, keeps packets' information available over time with a novel packet repetition strategy designed for this purpose. Furthermore, the effective cancellation of packets whose information has become obsolete reduces the channel load. SCF, a key mechanism to spread messages in sparse networks with limited node connectivity, has been enhanced by the introduction of distinct buffering and flush policies. These

policies allow GN to make smarter decisions on how to bridge temporally missing connectivity between network nodes. The foundation for all of GN's forwarding decisions is the LocT. Thus, the maintenance of LocTEs is of utter importance and deficits in this regard have been eliminated.

DCC gained two new interfaces with SCF valves and the resource allocation pattern for periodic services. These interfaces foster the cooperation between entities within an ITS-G5 station while still keeping the concerns of the layers isolated. The SCF valve allows DCC to interact with GN to drain its SCF packet buffer gradually without flooding the channel. DCC's interaction with periodic services enables applications to bring their messages with minimum delay on the radio medium. DCC-MAP allows stations to gain a more versatile view about the network's congestion at various locations, which is an improvement over DCC-MCO's uniform CBR perception. Earlier DCC mechanisms ignored the collaborative nature of multi-hop packets, with the effect that this traffic has been disadvantaged with rising channel congestion. The outlined forwarding budget enables transmitting these multi-hop packets even in congested environments by dedicating a share of the available channel capacity for common multi-hop forwarding. All these enhancements ultimately culminate in the novel Fallco DCC entity, which minimises latency for delay-sensitive messages without ruining the performance of delay-tolerant network traffic.

5 EVALUATION OF QOS MECHANISMS

Artery has become an extensive simulation model over the past years, with many configuration parameters. Aside from that, the third-party components Artery builds upon can also be customised in many ways. While it is desirable to have tools that can be adopted to individual needs in many ways, one can easily get caught up by the curse of dimensionality: Investigating just all combinations of enhancements outlined in the previous Chapter 4 would burst any reasonable page limit. Hence, only the most interesting variants can be studied hereafter. However, the tools and scenarios as introduced in Chapter 3 are shared with the research community for further studies.

5.1 BASIC SIMULATION CONFIGURATION

The V2X simulation framework Artery runs with OMNeT++ 5.6.2 and SUMO 1.6 in the following experiments. Source code of further dependencies, such as Vanetza and the INET Framework, is directly tracked in the Artery repository. The particular branch of the Artery repository used for the experiments, including the scenarios and their configurations will be published along with this thesis. As it would be depletive to describe every single parameter setting of such a complex simulation environment, the subsequent parameter description pin points essential aspects.

5.1.1 Vehicle Insertion Delays

It is common practice in SUMO that vehicles are inserted to the road network at integer seconds. For example, the modelled traffic demand of the LuST scenario starts at whole number seconds. Thus, the SUMO step length has to be reduced to 0.1 s so that vehicles generating CAMs at the maximum rate of 10 Hz are supplied with enough data.

However, if vehicles are inserted only at integer seconds and generate CAMs at a lower rate, the transmission time points are likely to get clustered artificially in the simulation. Such an unwanted clustering of messages is prevented by inserting new vehicles reported by SUMO with a random delay in Artery. This random delay is drawn from a uniform distribution ranging from 0 s to 1 s, the maximum message interval of the CA service.

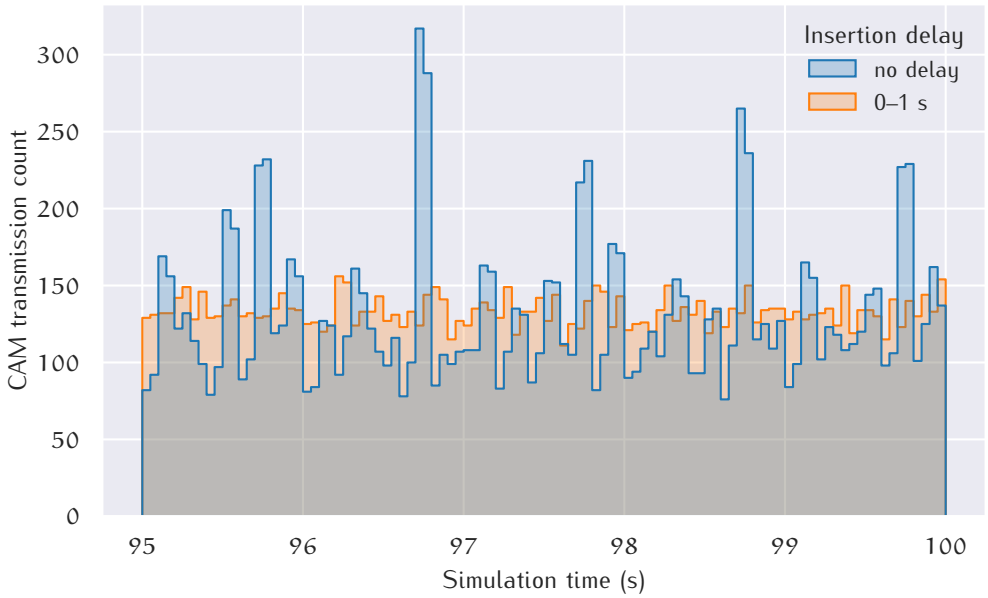


Figure 5.1: Distribution of CAM transmissions depending on vehicle insertion delay

Figure 5.1 shows the distribution of CAM transmissions in bins of 50 ms over time. It can be clearly seen that the delayed insertion, as described in the previous paragraph, distributes these transmissions evenly. Consequently, the channel load is not as concentrated at repeated time intervals as without delayed insertion. On the other hand, inserting vehicles instantly when they depart in SUMO causes significant, periodic peaks. These peaks are not a result of ITS-G5 itself but an unwanted side-effect of simulation that needs to be avoided.

5.1.2 Warmup Period

Because of the aforementioned delayed vehicle insertion, it is advisable to consider at least 1 s as warmup period of the simulation, *i.e.* not collecting data during this initial simulation phase. Ideally, all components of the simulation are in a steady state then.

Figure 5.2 shows the time until the CBRs averaged over all vehicles level off. Local CBR, either taking the mean of the last two measurements or the moving average calculation, are steady after a warmup period of 2 s. However, it takes about 3.4 s until DCC_NET propagates local CBRs through the network adequately.

The presented data in Fig. 5.2 has been gathered using the Griddy scenario, which features a constant number of vehicles, regular road network and homogenous traffic flows. These properties make it particularly suitable to assess the impact of the warmup period without other overlaying effects.

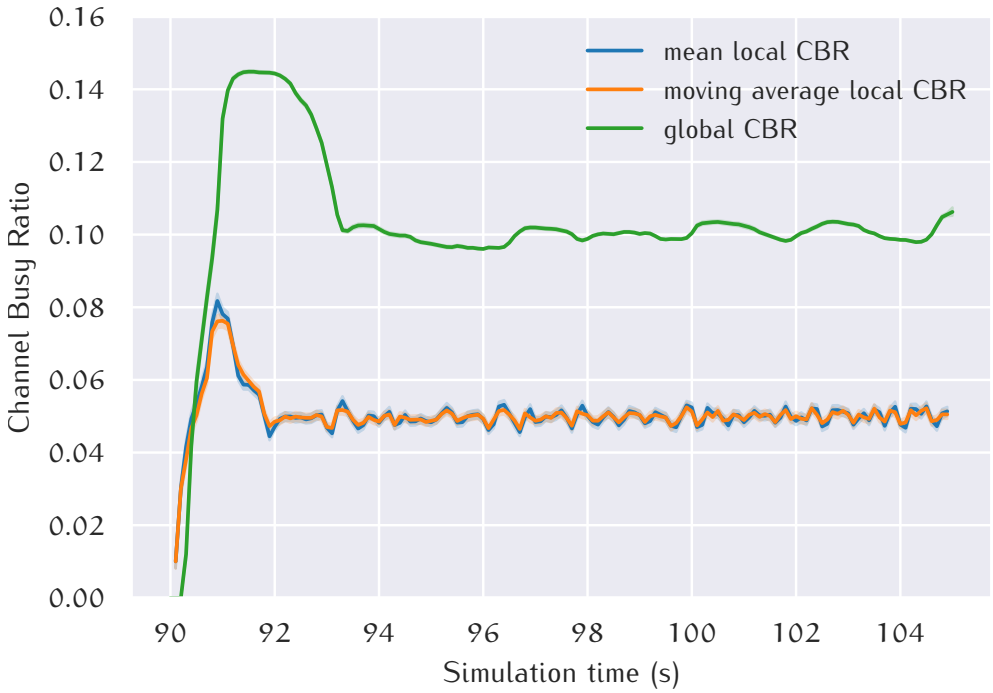


Figure 5.2: Steady state CBRs in simulation after warmup period

Consequently, the formed average CBRs of all vehicles reflects the individual CBRs quite accurately because the 95 % confidence intervals are narrow. These confidence intervals are plotted as semi-transparent error bands in Fig. 5.2.

At the beginning of the warmup period, when the vehicles are still getting inserted, CBR peaks can be observed. These peaks can be attributed to the additional GN beacon emitted once immediately by each new vehicle on top of the regular CAMs. This initial GN beacon is desirable to prime the LocTs of neighbours. As Fig. 5.3 shows, the number of recognised neighbours rises sharply for about 1.5 s. Further advancement of recognised neighbours on average depends heavily on the neighbour flag implementation as outlined in Section 4.1.2. The standardised behaviour, which does not explicitly reset this flag, causes further steady growth as the vehicles intermingle. The proposed behaviour, which resets the neighbour flags after an expiry duration, quickly converges to a steady state. The shorter the configured expiry duration, the faster the convergence level is reached.

Especially the simulation of scenarios with hundreds of vehicles can consume a lot of computation time, several hours per simulated second are common. Hence, the warmup period should be set as long as necessary but also as short as possible. Considering DCC_NET, the warmup period should be at least 3.4 s long. As has been argued in Section 4.1.2, a neighbour expiry of 3.75 s is the upper bound. In the interest of less computational load, the neighbour expiry is set to 2.0 s in this chapter. Vehicles, the predominant

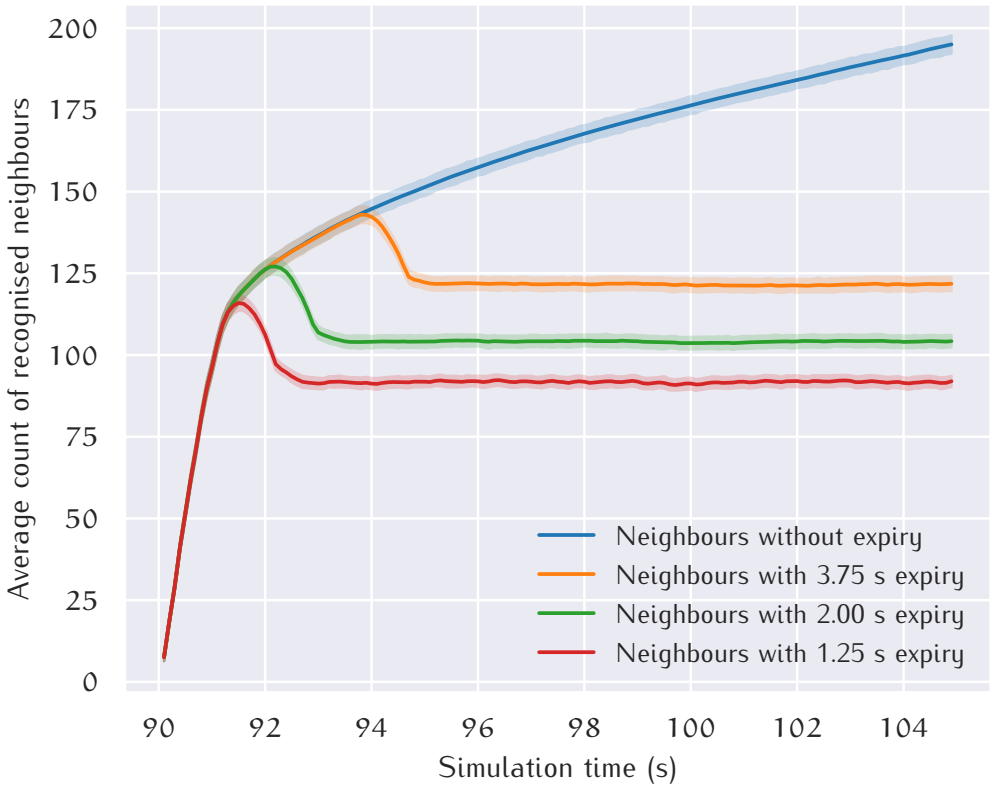


Figure 5.3: Steady-state LocTs in simulation after warmup period

station type in the following simulations, are equipped with the CA service and thus SHB packets are generated by each vehicle at least twice within 2 s. Thus, a shorter neighbour expiry can be justified.

5.1.3 Radio Propagation Settings

Every station in the evaluated scenarios is configured with identical ITS-G5 radios. While some radios may perform better than others concerning their receiver sensitivity for capturing packets in the field, these individual peculiarities are subordinated when evaluating the network's performance as a whole. For ease of reference, Table 5.1 summarises the essential parameter settings employed in the following.

As a general rule, packets will be transmitted with 6 Mbit/s at a power level of 100 mW. This data rate is mentioned as the standard option in [86] and [163]. The mentioned power level is also employed for CAM transmissions by the first series-production car equipped with ITS-G5 technology[1].

Section 3.2.1 presented two groups to model the propagation loss of radio signals: One group takes only the distance between radios into account to derive the attenuation using either a deterministic or a random distribution function. These functions may get calibrated for a particular environment through parameters, but the characteristics of the propagation loss are fixed

1 according to Wireshark trace files shared by vendor

Table 5.1: ITS-G5 radio settings

Parameter	Setting
Mode	IEEE 802.11p
Channel	180 at 5.9 GHz
Transmission power	20 dBm
Data rate	6 Mbit/s
Antenna	Omnidirectional 0 dBi
Antenna height	1.5 m above ground
Antenna polarisation	vertical
Noise floor	−98 dBm
Receiver sensitivity	−89 dBm
CCA signal threshold	−85 dBm
CCA noise threshold	−65 dBm
Capture threshold	6 dB
Minimum SNIR	2 dB
CBR threshold	−85 dBm
CBR reporting	asynchronously every 100 ms
Propagation loss	GEMV²

for the whole scenario. For example, the Nakagami fading model by Cheng et al. [43] is tuned for IVC in the 5.9 GHz band. The other group considers the actually present obstacles for each communication link, *e.g.* GEMV² evaluates the geometric properties of obstacles between transmitter and receiver.

On a large scale, the resulting distribution of reception ranges can be very similar between both groups. Figure 5.4 shows these distributions obtained with the Griddy scenario using GEMV² and Nakagami fading. Two variants of the Griddy scenario are simulated with GEMV², one without any buildings and the other with buildings arranged according to its 'plus islands' layout. Ranges obtained with the Nakagami fading model reside between the GEMV² variants. Thus, taken as a whole, an ITS application would experience comparable communication success rates relative to the communication distance.

While less complex models, such as the consulted Nakagami fading model, can imitate the distribution of communication success, they are unsuitable to reproduce the non-uniform spatial distribution of CBRs attributable to heterogeneous environments. Simulations focussing solely on the ITS application's perspective of communication may neglect this mismatch. However, because CBRs are a vital aspect for DCC, the evaluations in this chapter employ the GEMV² model. The used GEMV² settings, as listed in Table 5.2, are mainly taken from Boban, Barros and Tonguz [33] and the original reference implementation of GEMV² published along with Boban's PhD thesis [71].

For computational performance reasons, GEMV² employs range checks in a first step before calculating the attenuation in detail. These checks eliminate communication pairs, which are separated too far for successful

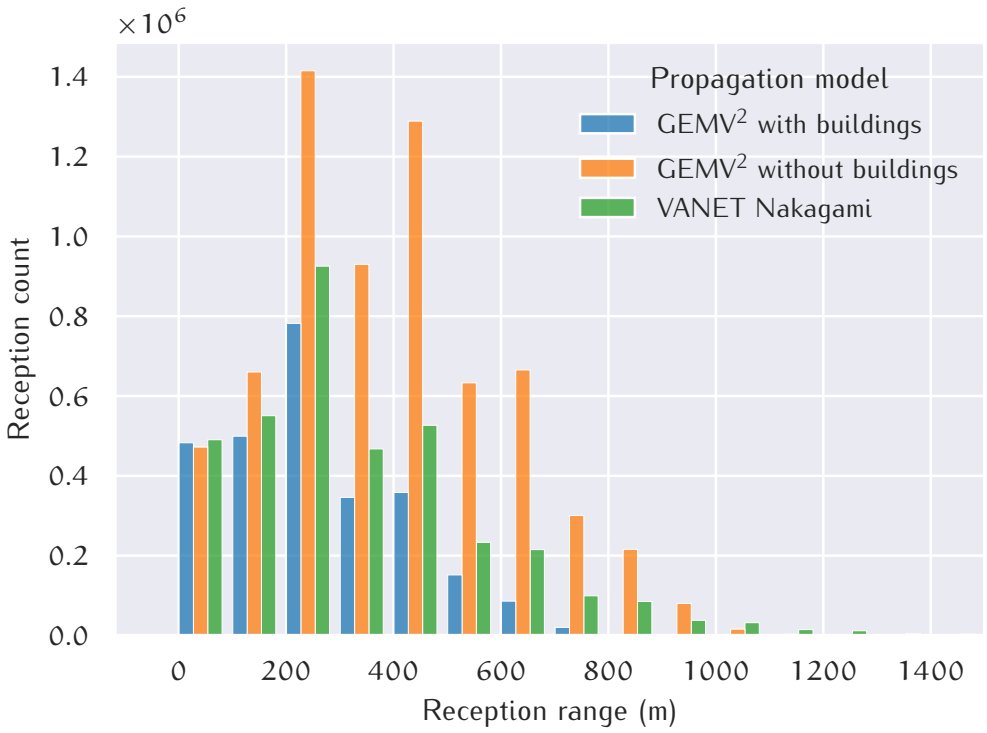


Figure 5.4: Receptions ranges with GEMV² and Nakagami propagation models

communication. For evaluations in this chapter, the range limit is set to 1500 m for LOS links and 1000 m for NLOS links. It is known from field experiments, that communication well beyond 1 km is highly unlikely. For comparison, Fig. 5.5 shows the number of receptions for 50 m-wide bins when a LOS-limit of 1500 m or 3000 m is applied. Latter limit is larger than the diameter of the employed Griddy scenario, *i.e.* no packet is dropped because of this range limit. If only the deterministic, large-scale fading model of GEMV² attenuates the signal, the range limit's effect is negligible. The amount of picked up interference can explain the slight differences in counted receptions per bin: More interfering noise can accumulate with a more extended range limit, which can cause CSMA/CA to switch from *idle* to *busy* channel state in rare cases. By adding the random small-scale variations, the effective reception range is extended; however, the applied range limit caps only very few receptions, which cannot even be displayed in Fig. 5.5 at this scale. Because the big picture is preserved with a limit of 1500 m, the benefit of considerably faster simulations runs – about factor 3 to 4 – justifies to ignore long-distance links.

5.1.4 Simulated ITS Applications

The available application models in Artery have already been presented in Section 3.2.5. Foremost, these are the Day One CA and DEN services, which are running on every vehicle equipped with ITS-G5. The CA service

Table 5.2: GEMV² radio propagation settings

Parameter	Setting
Small-scale variations	enabled
Static obstacles	SUMO *building* polygons
Vegetation obstacles	SUMO *forest*, *scrub*, *tree*, *tree_row*, *wood* polygons
LOS: no obstruction	
Model	two-ray interference
Relative permittivity	1.003
Cut-off range	1500 m
Small-scale variation (standard deviation)	3.3 dB to 5.2 dB
NLOSb: obstruction by buildings	
Model	full diffraction and reflection
Path loss exponent	2.9
Vehicle relative permittivity	6
Building relative permittivity	4.5
Cut-off range	1000 m
Small-scale variation (standard deviation)	0 dB to 6.8 dB
NLOSv: obstruction by vehicles	
Model	multi-knife-edge diffraction
Cut-off range	1000 m
Small-scale variation (standard deviation)	3.8 dB to 5.3 dB
NLOSf: obstruction by foliage	
Attenuation	2.33 dB/m
Cut-off range	1000 m
Small-scale variation (standard deviation)	4.1 dB to 6.8 dB

is the reason for most communication traffic due to its periodic messages. Events triggering the generation of DENMs are modelled with the aid of Artery's *storyboard*.

Two particular DEN use cases, traction loss and Electronic Emergency Brake Light (*EEBL*), generate non-emergency (DP1) and emergency (DP0) DENMs. With the aim to mimic the situation of suddenly forming black ice, the storyboard triggers with a preset likelihood that a vehicle loses traction within a one-second window after the simulation's warmup period. According to *Triggering Conditions and Data Quality - Adverse Weather Conditions* [165], the corresponding DENM is repeated every second after its first detection, and disseminated to vehicles within a 1 km perimeter.

Triggering Conditions and Data Quality - Dangerous Situation [166] describes a more critical use case, which is triggered by severely decelerating vehicles: If a vehicle decelerates by more than $-7\,\mathrm{m/s^2}$ for at least 0.5 s, an EEBL DENM

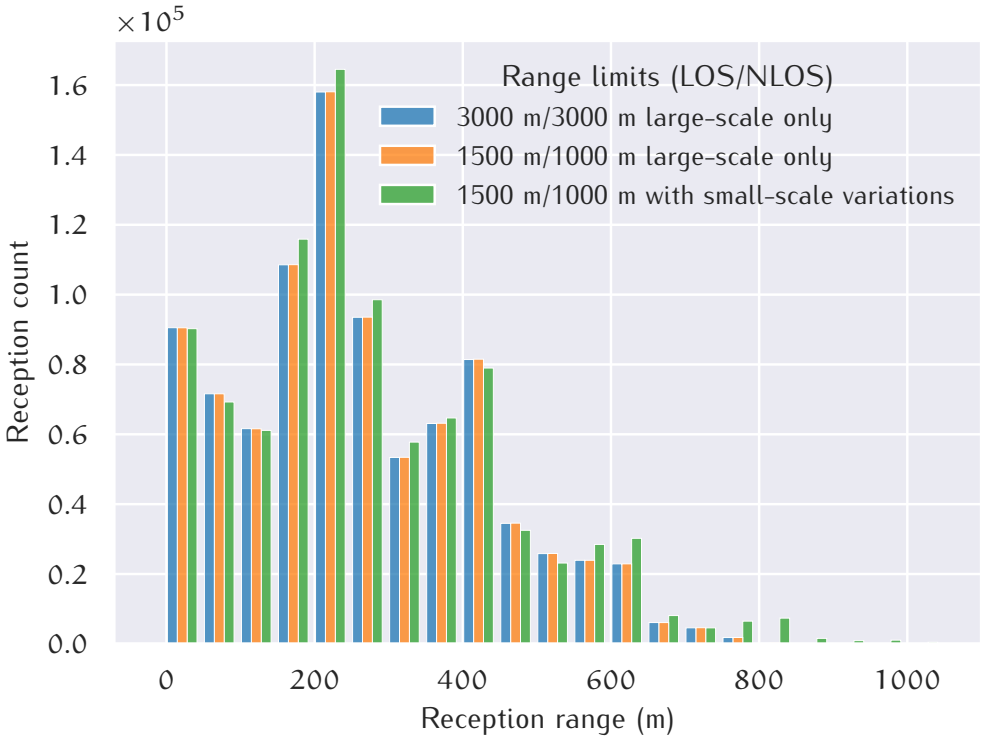


Figure 5.5: Impact of GEMV² range limits on reception histograms

is generated. These DENMs belong to the emergency TC and have only a short validity period of 2 s. Hence, these messages are not repeated and only addressed to vehicles within a distance of 500 m.

Half of the vehicles are also equipped with CP as another periodic service beside CA. Front and rear radar sensors with a range of 80 m and a field-of-view of 60° feed the CP services with perceived vehicles. Such an additional, periodic service helps to pinpoint issues of DCC with multiple running services, *i.e.* if channel resources have to be split up within a vehicle as well. Stations try to generate 10 CPMs per second, however, this rate may be throttled by DCC_FAC. Ongoing CP standardisation as per *TR 103 562* [175] outlines a plethora of redundancy mitigation rules, *i.e.* when to generate a CPM and also which containers to include then. For the evaluation purposes in this chapter, the strategy is kept simple by including all perceived objects. This CP, using a high rate and large CPMs, is thus suitable to add substantial packet traffic to the radio channel. DCC variants can then demonstrate their capability to handle more than one periodic service.

Last but not least, RSUs run infrastructure services at fixed positions. RSUs located at traffic lights emit SPATEMs and MAPEMs as those messages are typical for this type of intersections. These infrastructure messages are disseminated within a radius of 400 m once per second, in line with *TS 103 301* [144].

5.1.5 Security

Security of ITS-G5 is enabled in the following simulations. In particular, the signing policies are adhered to, and the overhead due to longer packets is considered. To keep computational load low, packets are never validated and thus the deferred signature calculation is never triggered. Conceptionally, this approach is a slightly updated version of [7].

In summary, CAMs usually convey just a certificate digest but at least once per second the full certificate. The full certificate is enforced for all DENMs and all other traffic without explicit policy. As an exception to this rule, CPMs always include the shorter digest because the CP service is only used in conjunction with the CA service.

5.1.6 Congestion Control

To keep the magnitude of DCC variants low, three flavours of congestion control are considered. All three flavours share similar configuration settings where possible, *i.e.* as far as the particular characteristic allows such a configuration.

1. reactive DCC profiled by BSP [163]

2. adaptive DCC based on LIMERIC [143]

3. Fallco DCC as outlined in Section 4.2.5

The BSP explicitly demands to take the mean of the last two CBR measurements as input to trigger transitions of its state machine. On the other hand, both DCC variants employing LIMERIC calculate a moving average of the global, network-based CBR. Local raw CBR measurements are processed by DCC_NET and only its output is fed into LIMERIC.

All three variants share four queues at DCC_ACC with capacities for two packets each. Furthermore, DPo bursts for emergency cases are allowed by all of them. Latter two variants incorporate the LIMERIC algorithm, which is implemented and configured according to *TS 102 687* [143]. The only deviation from the original specification is the addition of the dual-α enhancement, as discussed in Section 2.4.2. Fallo DCC, the newcomer among these DCC variants, is further studied with static and elastic forwarding budget.

5.2 RURAL ENVIRONMENT

DekiNet2 represents a rural environment with many country roads and villages but also a motorway dissecting the countryside. The road topology and environmental conditions such as vegetation of *DekiNet2* have been detailed in Section 3.1 on Pages 85 to 88. Data from this scenario is collected in the simulation time window from 900 s to 905 s when over 2300 vehicles are running. As outlined in Section 5.1.4, all of them are equipped with CA and DEN as basic services, half of them also with CP services. The DEN traction loss use case is triggered with a probability of 50 % by each vehicle. Despite *DekiNet2*'s large area size, this map is not well-suited for traffic light services because it features only two traffic lights at all.

Further GBC traffic, aside from DENM, is generated by six RSUs emitting RTCM messages for augmented positioning via GNSS. Figure 5.6 highlights the placement of RSUs operating the GNSS augmentation service with concentric destination areas with diameters of 3.5 km. The size of those circular destination areas is thus at the upper end with respect to GN's `itsGnMaxGeoAreaSize` = 10 km² limit. Thus, each RSU sits in the middle of its respective destination area, coloured in green. Motorway A9 crosses three (A, B and D) of six coverage areas, one other (C) is in the vicinity to the motorway, whereas the remaining two (E and F) are in the countryside near to communities of different size.

5.2.1 Coverage and Latency of GeoBroadcasts

Figure 5.7 depicts the coverage over time of the six RSUs, which are placed according to the map shown in Fig. 5.6. Within the simulated time window, each RSU generates five RTCM messages in total. Their start time points of dissemination are not synchronised among the RSUs. The coverage ratio, *i.e.* the number of distinct, actual receivers divided by the number of stations within the addressed area, varies a lot. The placement of the RSUs is crucial for wide dissemination, irrespective of the employed DCC mechanism. It turns out that RSU A, which is located in the middle of a village, is sometimes unable to achieve any notable coverage with any DCC variant. No vehicle is close enough to the RSU in these cases to act as the first forwarder in the routing chain. Other RSUs are placed more favourable, but reactive DCC achieves only low coverage ratios in most constellations. Adaptive DCC is an improvement over reactive DCC, but cannot always compete with the coverage ratios by Fallco DCC. In the comparatively crowded destination area of RSU F, Fallco DCC suffers less from dropped forwarding packets and

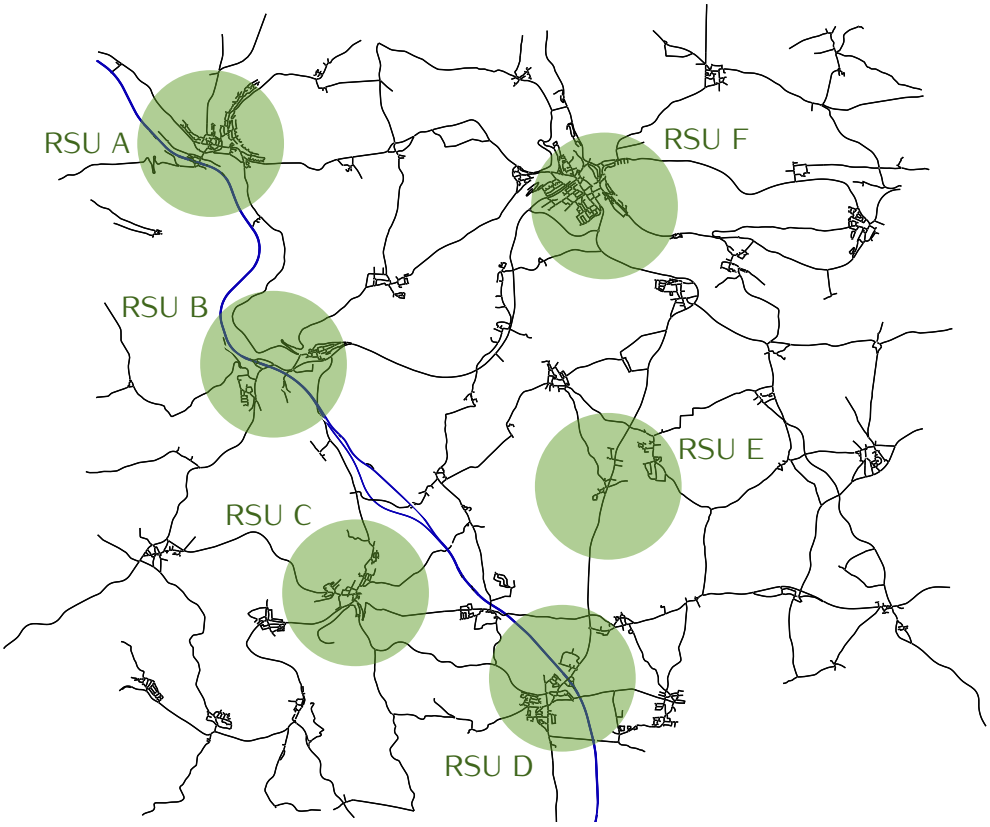


Figure 5.6: Placement of RSUs's destination areas on DekiNet2 map

is thus rewarded by higher coverage ratios. Especially the elastic forwarding budget pays off in the rural DekiNet2 scenario as it offers more space for forwarding transmissions.

A look on the dissemination events occurring with each DCC variants, plotted in Fig. 5.8, reveals why the usage of Fallco tends to result in better coverage. The number of forwarding operations by GN accumulated in 100 ms bins is quite similar among the studied variants. Slight differences exist as forwarding by later hops depends on the successful dissemination to earlier hops. Reactive DCC suffers from a huge amount of packet drops by DCC_ACC. Hence, the majority of packet forwarding transmissions demanded by GN is actually never transmitted on the radio. Adaptive DCC drops fewer packets, and thus its coverage ratios are better compared to reactive DCC, as reflected by Fig. 5.7. However, about one out of four forwarding transmissions is still dropped.

Packet drop at DCC_ACC is not an issue observed with both Fallco variants, which employ the forwarding budget introduced in Section 4.2.4. Roughly equivalent to packet drop is the depletion of forwarding budget in these cases. In fact, on some occasions, a station has to refrain forwarding because its budget limit is reached. The frequency of these events is two magnitudes below the packet drop by reactive and adaptive DCC, though. Since the

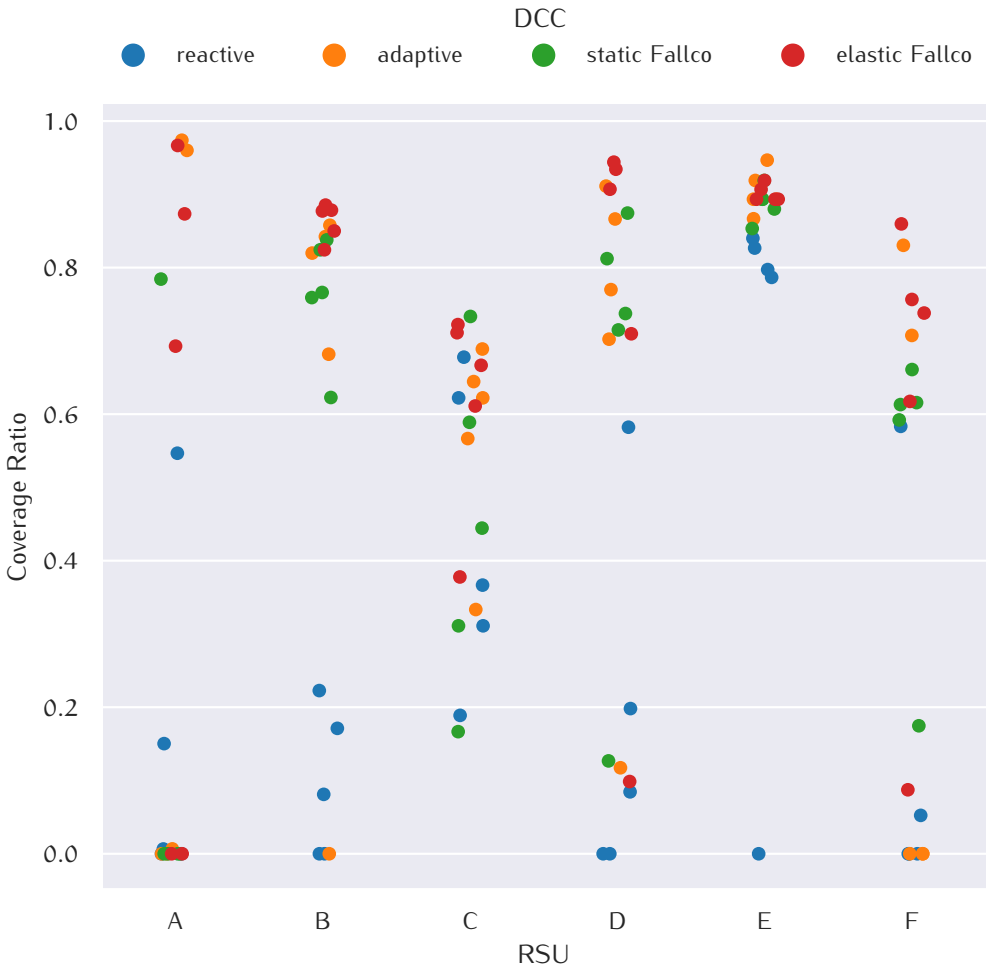


Figure 5.7: Coverage by RTCM RSU stations in DekiNet2 scenario

overall traffic density and thus also the network congestion is not extreme in this scenario, the elastic expansion of the forwarding budget gives the elastic Fallco variant a slight advantage.

Another advantage of Fallco comes to light by investigating the RTCM latencies, *i.e.* the duration until a station receives an original message for the very first time. Duplicate receptions are thus neglected. One can spot at the low-latency end of Fig. 5.9, that CBF is delayed most by reactive DCC. Forwarding under the adaptive DCC regime is already quicker; however, both Fallco DCCs show the highest amount of low-latency forwarding transmissions. This observation also indicates unhampered CBF operation, *i.e.* the overhearing algorithm is not rendered inoperative by Fallco DCC. In accordance with Fig. 5.7, the availability of forwarding budgets with Fallco DCC aids to reach considerably more addressed stations overall. Queuing of forwarding transmissions, as it happens in particular with reactive DCC and to some degree also with adaptive DCC, is not only prone to excessive packet drops but also increases latency for those forwarding packets finally transmitted at all.

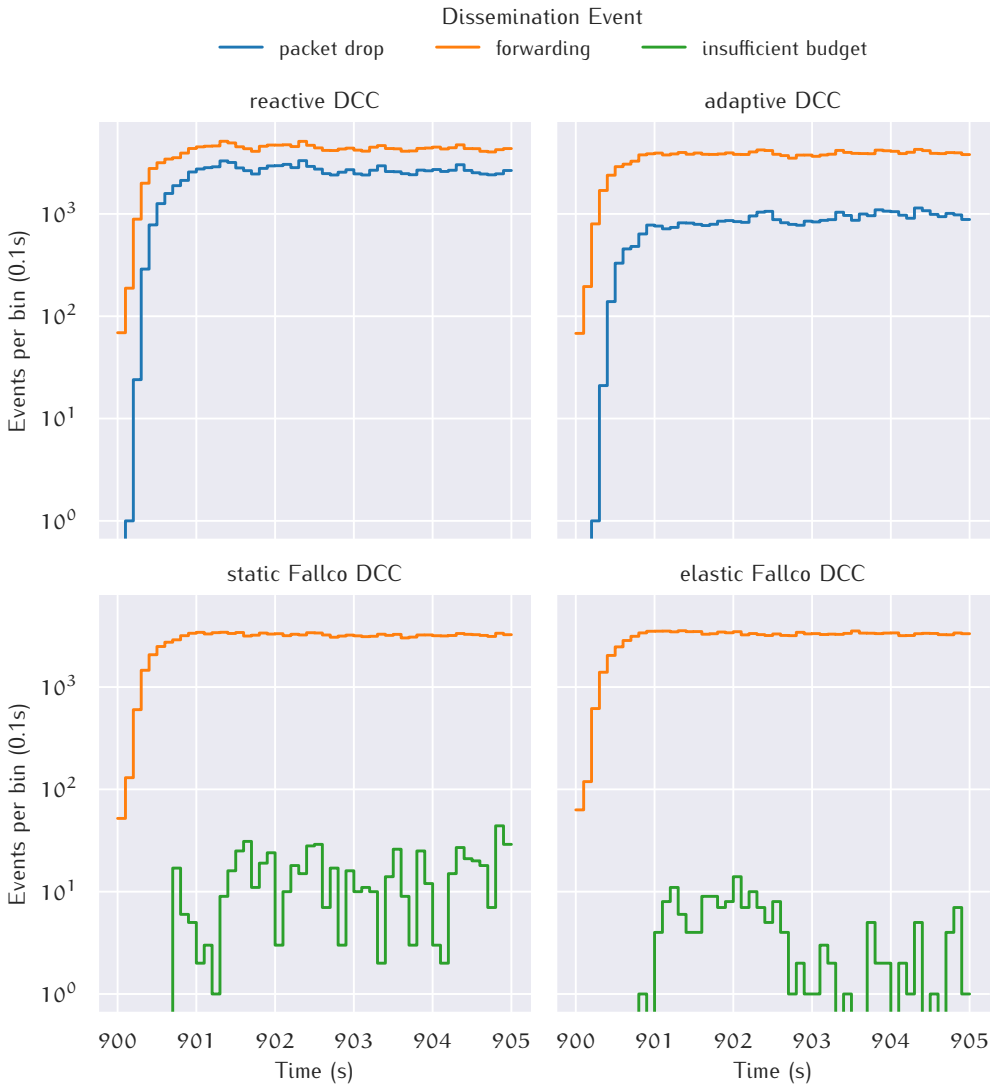


Figure 5.8: Dissemination events DekiNet2 scenario

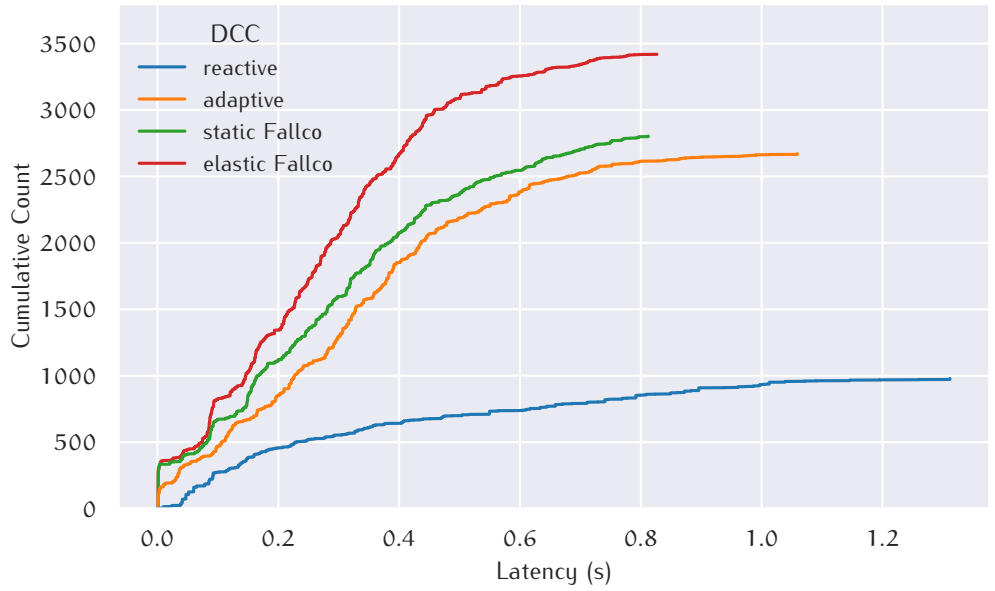
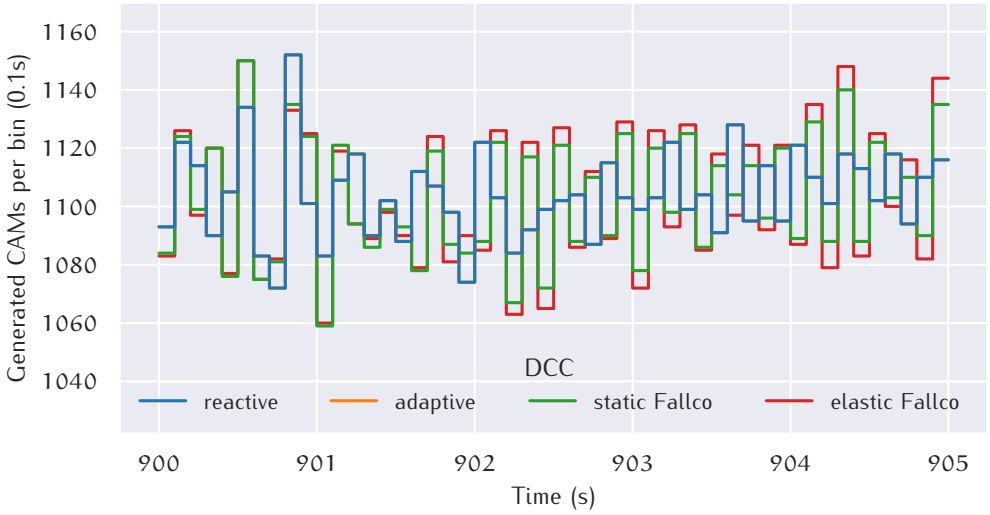


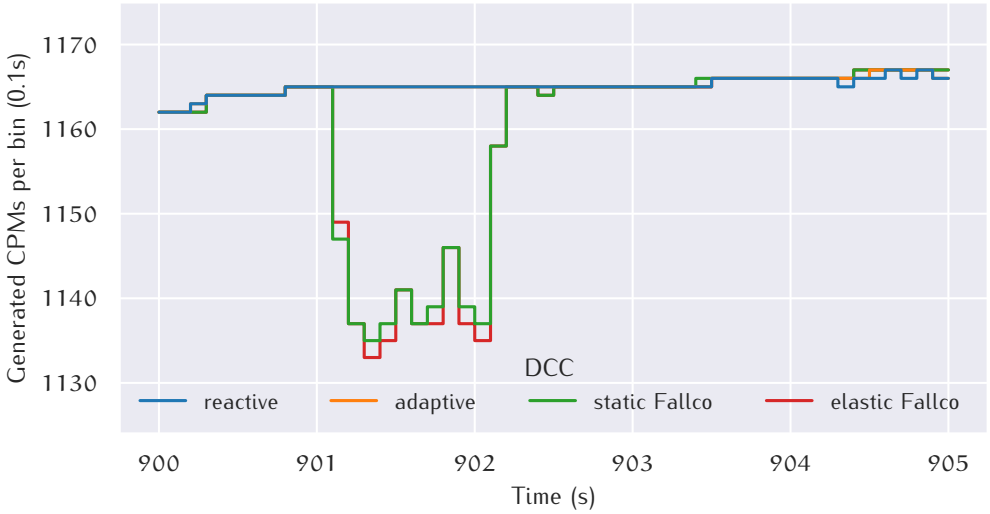
Figure 5.9: Cumulative distribution of RTCMs latencies in DekiNet2 scenario

5.2.2 Periodic Messages

Beside the forwarding budget, Fallco further features the periodic service interface, which shall allow practically immediate transmission of newly generated messages. As long as sufficient channel resources are available, this interface is expected to generate a similar amount of messages as the respective service would do without DCC constraints. Figure 5.10 shows the amount of CAMs and CPMs generated by all vehicles aggregated into bins of 100 ms. Over the entire period of 5 s, reactive and adaptive DCC generate 55 211 CAMs respectively. They both throttle their generation rate by CA's triggering conditions and DCC_ACC's TRC constraint. As the plotted curves for reactive and adaptive DCC are congruent, the TRC behaviour makes no difference in this rural environment.



(a) Sum of generated CAMs every 100 ms



(b) Sum of generated CPMs every 100 ms

Figure 5.10: Generation of periodic messages

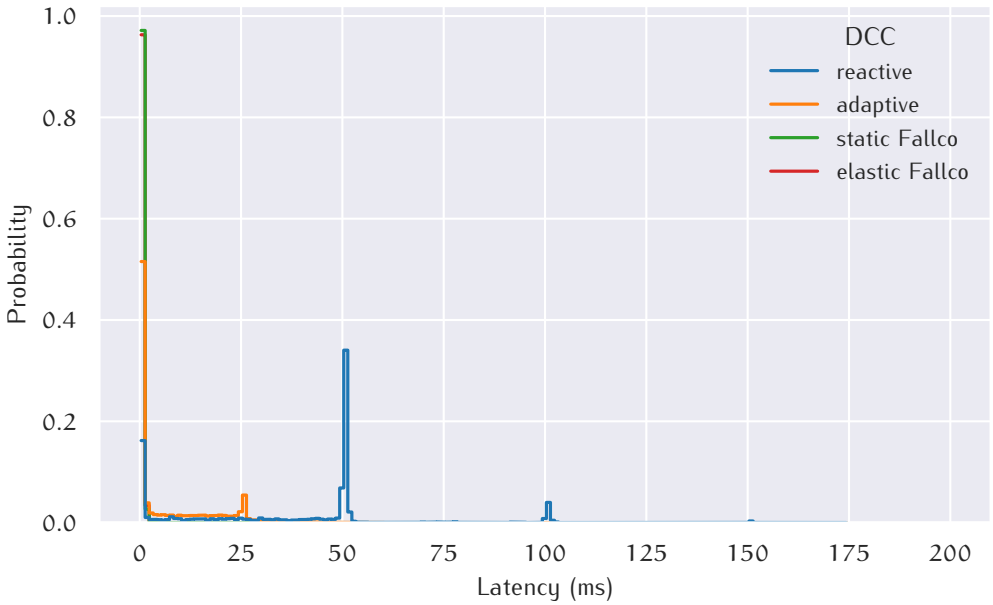
The inversion of message triggering enabled by Fallco – DCC notifies CA at time points where the service is allowed to generate a message – results in slightly different plots. However, the sum of generated CAMs remains just a little bit below traditional TRC. See Table 5.4 for a tabular summary of message occurrences with each DCC variant. Due to the presence of DP1 DENMs emitted by the traction loss use case, it is expected that these numbers are smaller: DCC shall not offer a periodic service a transmission opportunity as long as higher priority DENM needs to be served.

Likewise, CPMs are stoically generated at a fixed rate of 10 Hz for reactive and adaptive DCC. The difference of messages per bin can be attributed to the changing number of vehicles in the network. Also with the Fallco setup, the CP service aims to generate a message every 100 ms. However, Fallco DCC may not always fulfil CP's desired message rate as visible from 901.1 s to 902.2 s in Fig. 5.10b: About 30 CPMs less are generate in this time interval, *i.e.* DCC offered fewer transmission opportunities to periodic services. The scheduling policy for the periodic services aims to distribute airtime equally among services of the same DP. Since the CA service may waive a transmission opportunity if its triggering conditions are not met, the tracked channel occupancy by CAMs is likely below the occupancy by CPMs. Thus, the effective CAM rate remains stable while the CPM rate gets throttled as the occurrence of transmission opportunities temporarily drops. Only a small subset of the over 1160 vehicles equipped with CP services is affected by this rate reduction, though. Data not shown here also confirms that multiple vehicles reduce their CPM rate slightly, *i.e.* at most one CPM less over 5 s, but no single vehicle reduces its rate drastically.

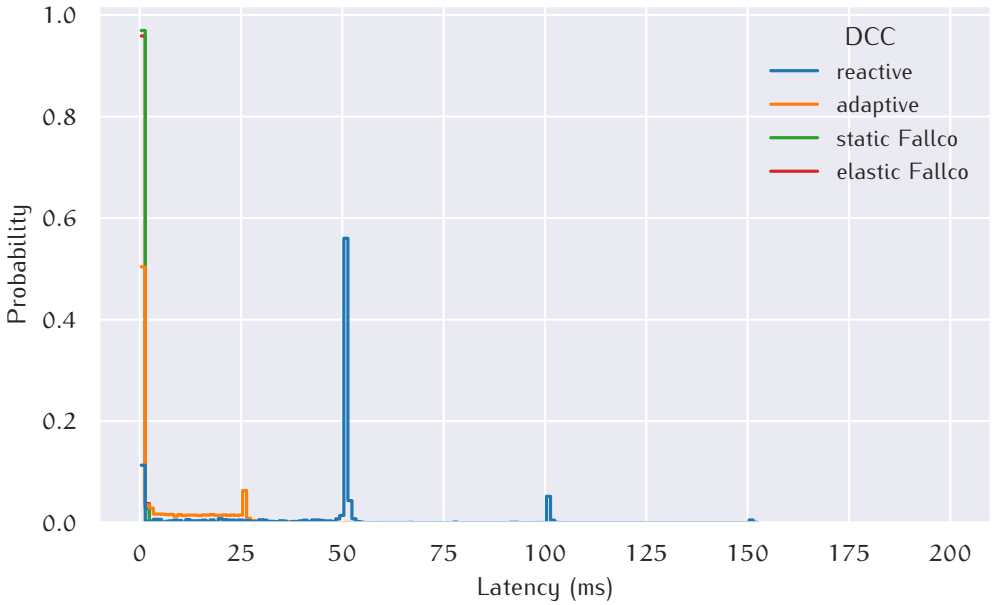
Histograms of the latencies observed with periodic messages are plotted in Fig. 5.11. These latencies account for the overall latency at the service layer, *i.e.* account for propagation and MAC' channel access delay as well as sojourn time in DCC queues. Both employed periodic services suffer from unnecessary latencies, often around multiples of 25 ms. Since a minimal transmission interval of 25 ms is enforced by DCC_ACC, these latency values are likely when the queues fill up. This effect is more pronounced with reactive DCC than with adaptive DCC, which hints at the rather coarse TRC by reactive DCC's state machine. The goal of the novel periodic service interface offered by Fallco DCC is achieved, though, as latencies never exceed 6 ms. Complementary to the shown plots, latency quantiles are listed in Table 5.3.

5.2.3 Impact of Forwarding Budgets

Plots in Fig. 5.8 suggest that the simulated networks with Fallco DCC variants exhibit higher CBRs due to the larger amount of packet forwarding transmissions. Figure 5.12 shows the empirical cumulative distribution



(a) CAM latency probabilities in 1 ms bins



(b) CPM latency probabilities in 1 ms bins

Figure 5.11: Latencies of periodic messages

Table 5.3: Latency quantiles of periodic messages

DCC variant	CAM latency (ms)			CPM latency (ms)		
	90 %	99 %	100 %	90 %	99 %	100 %
Reactive	51.6	101.6	174.3	51.8	101.6	152.2
Adaptive	25.3	42.2	89.0	25.3	32.9	98.5
Static Fallco	0.9	1.7	3.9	0.9	1.8	4.8
Elastic Fallco	1.0	1.8	4.4	1.0	1.9	6.0

Table 5.4: Message occurrences in DekiNet2

DCC variant	Generated			Received (distinct)		
	CAMs	CPMs	DENMs	CAMs	CPMs	DENMs
Reactive	55 211	58 253	1146	768 555	382 065	61 978
Adaptive	55 211	58 256	1146	743 888	373 436	68 948
Static Fallco	55 137	57 995	1146	777 168	385 282	68 651
Elastic Fallco	55 148	57 986	1146	764 583	379 147	69 821

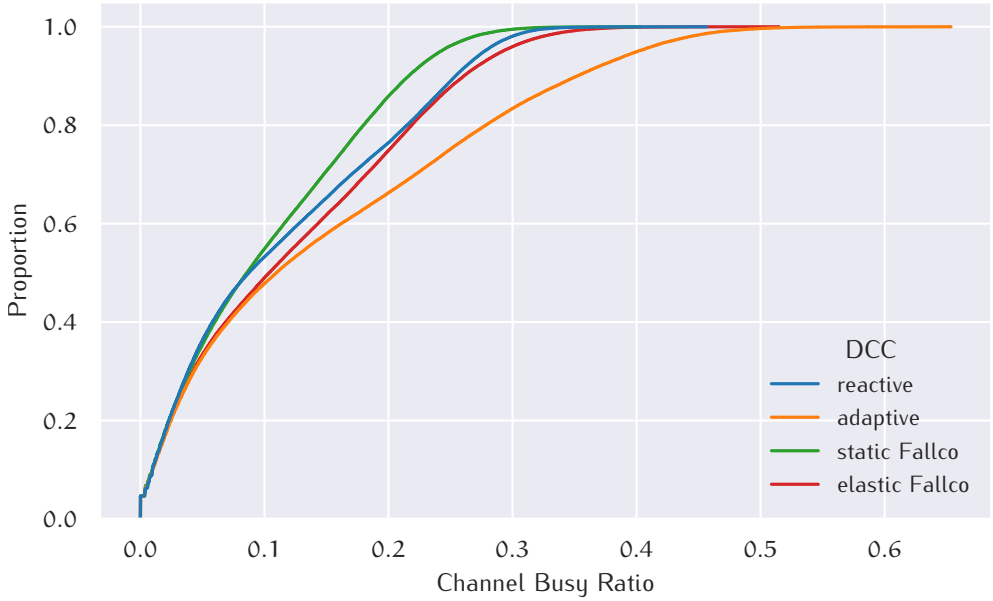


Figure 5.12: Cumulative distribution of CBRs

functions of the measured local CBRs collected between 900 s and 905 s. As expected, the elastic forwarding budget leads to a higher average CBR compared to its static forwarding budget counterpart. At the same time, the more at forwarding capacity leads results that 1800 more distinct stations receive one of the disseminated DENMs. Largest channel congestion is observed with adaptive DCC. Despite the broken CBF overhearing – as outlined earlier with the aid of Fig. 5.8 – adaptive DCC is also capable of disseminating DENMs to a large audience. Nevertheless, the elastic Fallco variant excels in summary: Periodic messages are generated without significant hindrance compared to reactive and adaptive DCC and its DENM dissemination performs best.

The effect of elastic forwarding budgets can be seen in Fig. 5.13. In a rural environment as DekiNet2, channel utilisation is usually not at its limit. In fact, with Fallco DCC variants the CBR is never above 50 % for any station at any time. This chance is taken by the elastic forwarding budget to reserve otherwise unused channel resources for packet forwarding. Consequently, about 98 % of all stations have still half of 'their' forwarding budget left.

As the forwarding budget is a concept that is consumed and monitored jointly by stations, no harsh difference should be observable among nearby stations. Otherwise, channel resources would be distributed unfairly, and

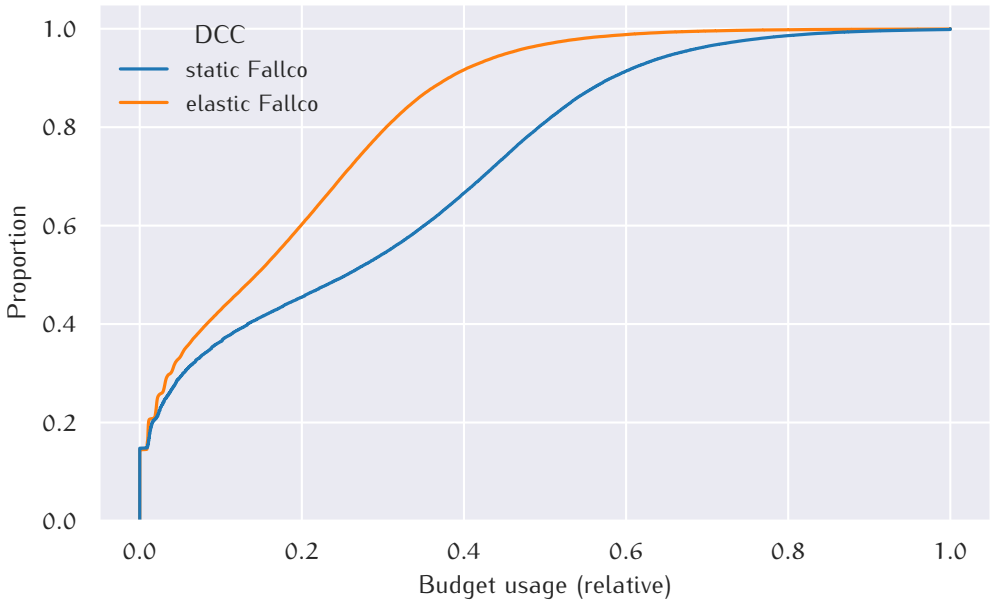
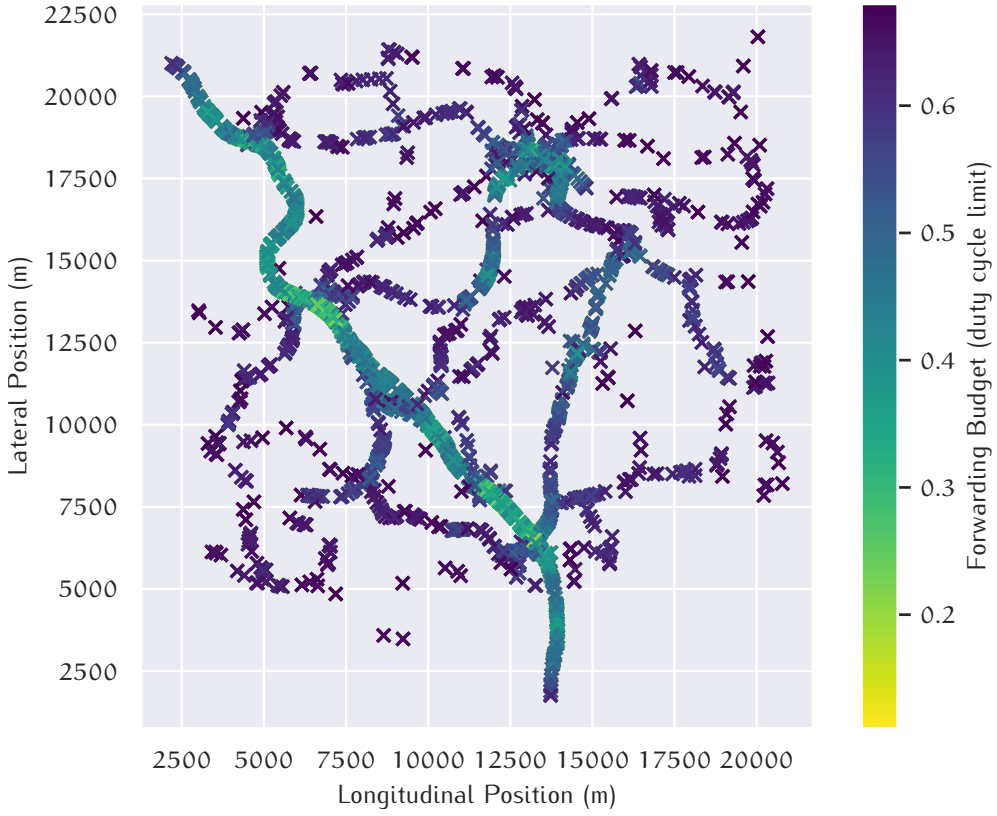


Figure 5.13: Usage of assigned forwarding budgets

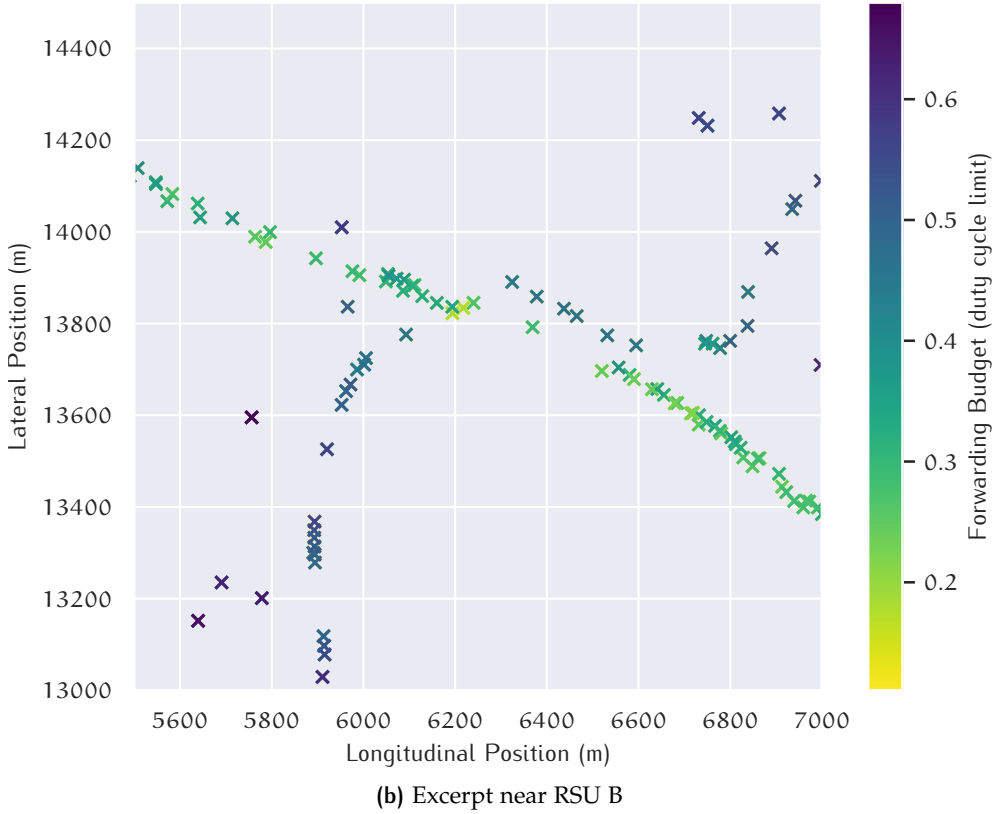
forwarding algorithms may suffer from side-effects occurring when only a subset of stations is operating at their budget limits. Two aspects are relevant at this point: First, the anticipated budget in terms of the allowed duty cycle should be similar for neighbouring stations. Second, the monitored forwarding budget usage in terms of channel occupancy by all nearby forwarding transmissions shall be evenly balanced.

The forwarding budgets determined by each vehicle within the simulation time window 901.9 s to 902.0 s are plotted in Fig. 5.14 for the elastic budget variant. In sparse areas with very low channel utilisation, the forwarding budget effectively expands to fill the gap up to the target CBR. In areas with higher vehicle densities and within the coverage areas of RSUs, the forwarding budget shrinks as desired considerably. As can be seen in Fig. 5.12, the network hardly operates at the target CBR though and thus even at the yellowish hot spots a budget of roughly a fifth of airtime is allocated. Distribution of forwarding budgets in Fig. 5.14 is not perfectly smooth but also not random, though. Stations report their budgets when they collect a new CBR measurement, *i.e.* the reported budgets do not belong to a single instance in time. Hence, some slight deviations can be explained by these shifted observation time frames.

Spatial distribution of monitored forwarding occupancy shows a similar picture in Fig. 5.15. Only a few forwardings occur in remote areas if a vehicle detected a traction loss within the plotted 0.1 s time window. Tracked channel occupancy within the dissemination area of RSUs is more pronounced clearly. Again, neighbouring stations report very similar duty cycles used for packet forwarding, which indicates an operational, common view on the consumption of the forwarding budget. Another observation is that vehicles



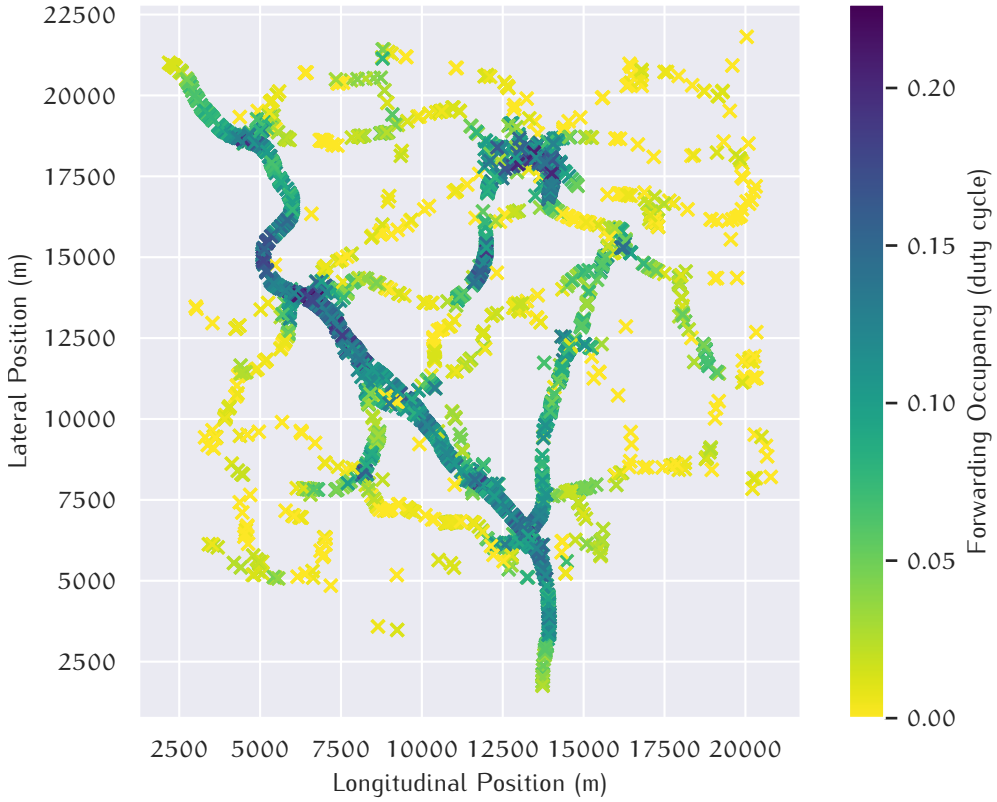
(a) Whole map



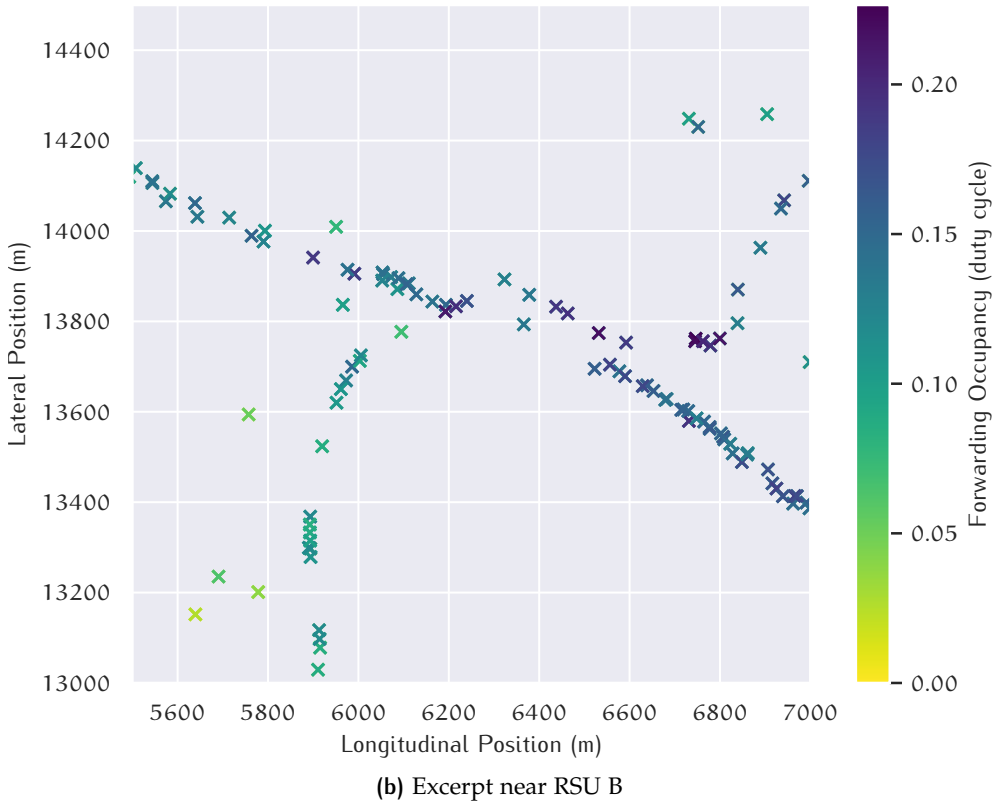
(b) Excerpt near RSU B

Figure 5.14: Forwarding budgets reported in time interval 901.9 s to 902.0 s

on neighbouring roads, *e.g.* parallel to the motorway, often perceive quite different occupancies though they are just a few hundreds meter apart. The vegetation in the form of wooded areas attenuates the signal considerably in the DekiNet2 environment.



(a) Whole map



(b) Excerpt near RSU B

Figure 5.15: Forwarding occupancy reported in time interval 901.9 s to 902.0 s

5.2.4 Effect of DCC Information Sharing

So far, DCC used simply local CBR measurements. At this point, DCC information sharing is added to the setups in two flavours. One is the standardised CBR piggybacking scheme using the maximum of local, one-hop, and two-hop as global CBR. The other is the directional CBR scheme presented in Section 4.2.1. Former is referred to as 'two-hop-max', latter as 'cbr-map' variant of DCC_NET information sharing.

Irrespective of the combined DCC_ACC mechanism, the CBR input with information sharing is distinctively higher compared to the local-only CBR input. The corresponding plots in Fig. 5.16 further highlight that granularity of CBRs is getting coarser from local-only, two-hop-max to cbr-map. Despite the differences in granularity, cbr-map is able to follow two-hop-max closely though the aggregation method differs substantially. As listed in Table 5.5, dissemination of DENMs suffers from the higher CBR inputs under the two-hop-max and cbr-map schemes. Fallco with its separate forwarding budget, which is generous in the relatively sparse DekiNet2 scenario, is not affected by the employed CBR information sharing, though.

5.2.5 RTCM Dissemination with SCF and Glow Forwarding

Mechanisms such as SCF and Glow Forwarding, first introduced in Section 4.1.3, are designed to overcome temporarily interrupted connectivity in sparse networks. Thus, the dissemination of RTCM messages may benefit from these mechanisms. In the previous experiments, each RSU generated a RTCM message every second, *i.e.* five in total over the investigated simulation time window. The dissemination of these GBC packets has been CBF without SCF nor Glow Forwarding. In addition to the coverage achieved by DCC variants plotted in Fig. 5.7, the coverage achieved under various GBC

Table 5.5: Message occurrences with CBR information sharing

DCC	CBR scheme	CPM transmissions	CPM receptions	DENM receptions
Reactive	local-only	58 253	382 065	61 978
	two-hop-max	58 256	355 496	60 444
	cbr-map	58 256	357 240	60 797
Adaptive	local-only	58 256	373 436	68 948
	two-hop-max	58 256	376 602	68 710
	cbr-map	58 236	375 594	68 200
Fallco	local-only	57 986	379 147	69 821
	two-hop-max	57 998	381 441	69 689
	cbr-map	57 985	380 736	69 560

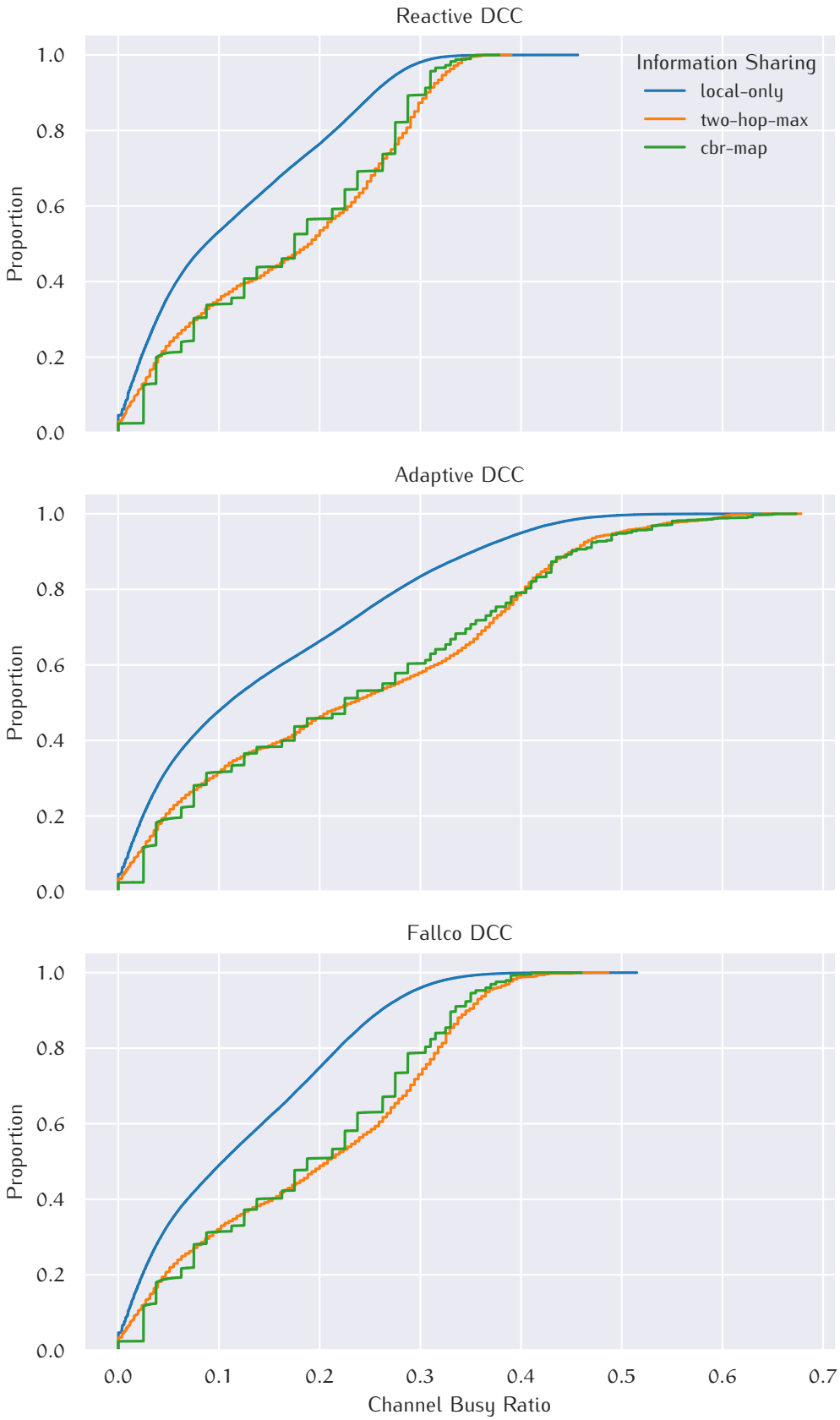


Figure 5.16: Cumulative CBR distribution under various information sharing schemes

dissemination variants is shown in Fig. 5.17. For all these GBC variants the elastic Fallco DCC has been employed. If Glow Forwarding is enabled, the repetition scheme built into Glow Forwarding replaces explicit repetitions by the RTCM service. Thus, RSUs generate only a single RTCM message each with a Glow Rate of 1 s set. Glow Forwarding results are consequently depicted by a single dot per RSU, while five individual dots represent the RTCM services' repetitions, which are distinct GBC packets with differing sequence numbers.

In contrast to Fig. 5.7, the small-scale variations of GEMV² have been disabled for Fig. 5.17, so a comparison between the dissemination patterns is possible under stable, deterministic conditions. The difficult network topology of RSU A is precarious: Due to the small, random variations in signal attenuation, some coverage has still been possible in the setup shown by Fig. 5.7. Without those occasional 'signal boosts', the connectivity is in fact interrupted, and no coverage is achieved at all. No GBC dissemination variant can make any difference without network connectivity. If connectivity among vehicles exists, however, the Glow Forwarding dissemination combined with SCF usually excels the traditional packet repetitions. Only with RSU C, a single packet repetition achieves a higher coverage then the Glow Forwarding with SCF tandem.

The timing perspective behind RTCM coverage is detailed by the plots of Fig. 5.18. These plots show the number of reached receivers by each GBC dissemination variant. Each receiving station is accounted when it has the received the RSU's message for the first time, *i.e.* all repetitions are treated equally. Looking at the plots using simple packet repetitions, it is hard to argue that SCF should be always enabled. While more stations are covered by RSU C's first dissemination phase with SCF, the repetitions without SCF cover ultimately more distinct stations. RSU E's plots suggest that repetitions with SCF have a minor advantage over plain repetitions. All other cases, except RSU A, put forward a convincing case to omit SCF. Then again, SCF seems to a beneficial addition to Glow Forwarding because it outperforms plain Glow Forwarding in most cases. With RSUs D and E, plain Glow Forwarding is initially faster in the dissemination process; however, this only a real advantage with time-critical information, but not with RTCM messages.

The difference on the network's overall congestion is negligible among the GBC dissemination variants. Their cumulative distribution plots are congruent essentially and thus not shown here. Impact on CBRs may be more pronounced in a dense, urban environment, though.

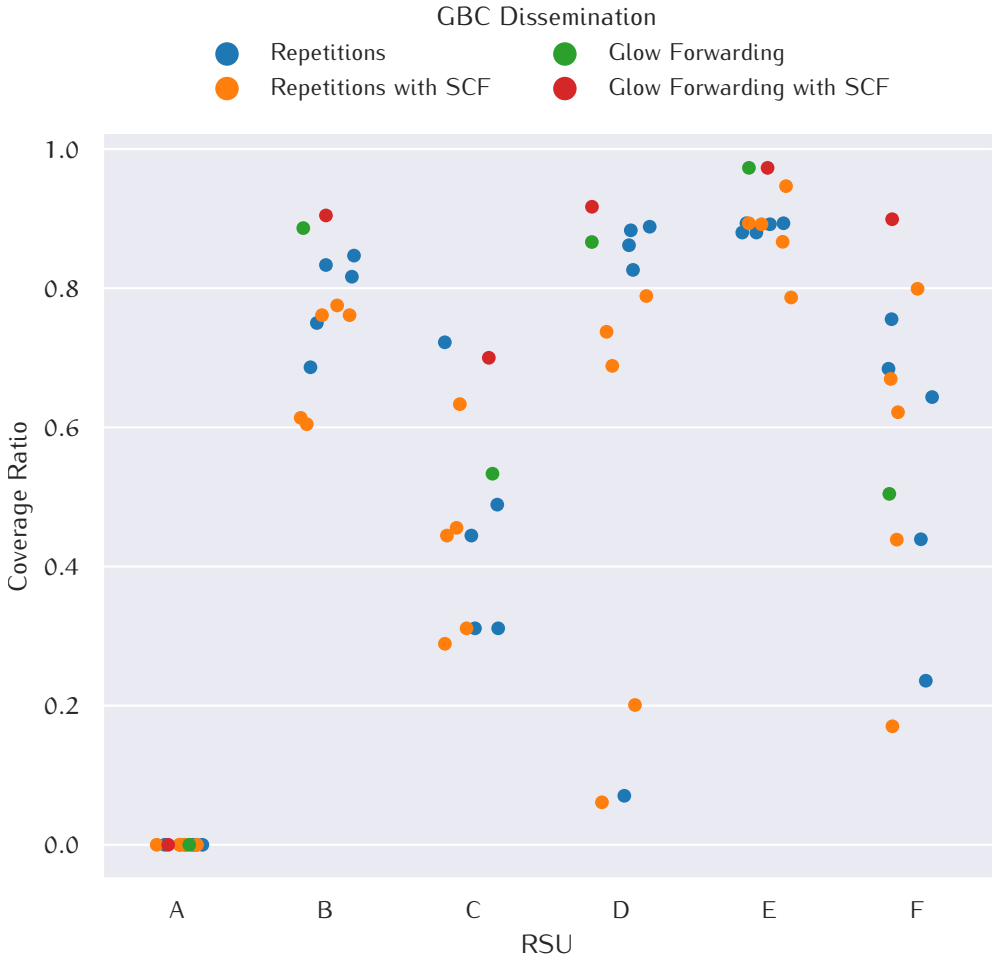


Figure 5.17: Coverage by RTCM RSU stations with SCF and Glow Forwarding in DekiNet2 scenario

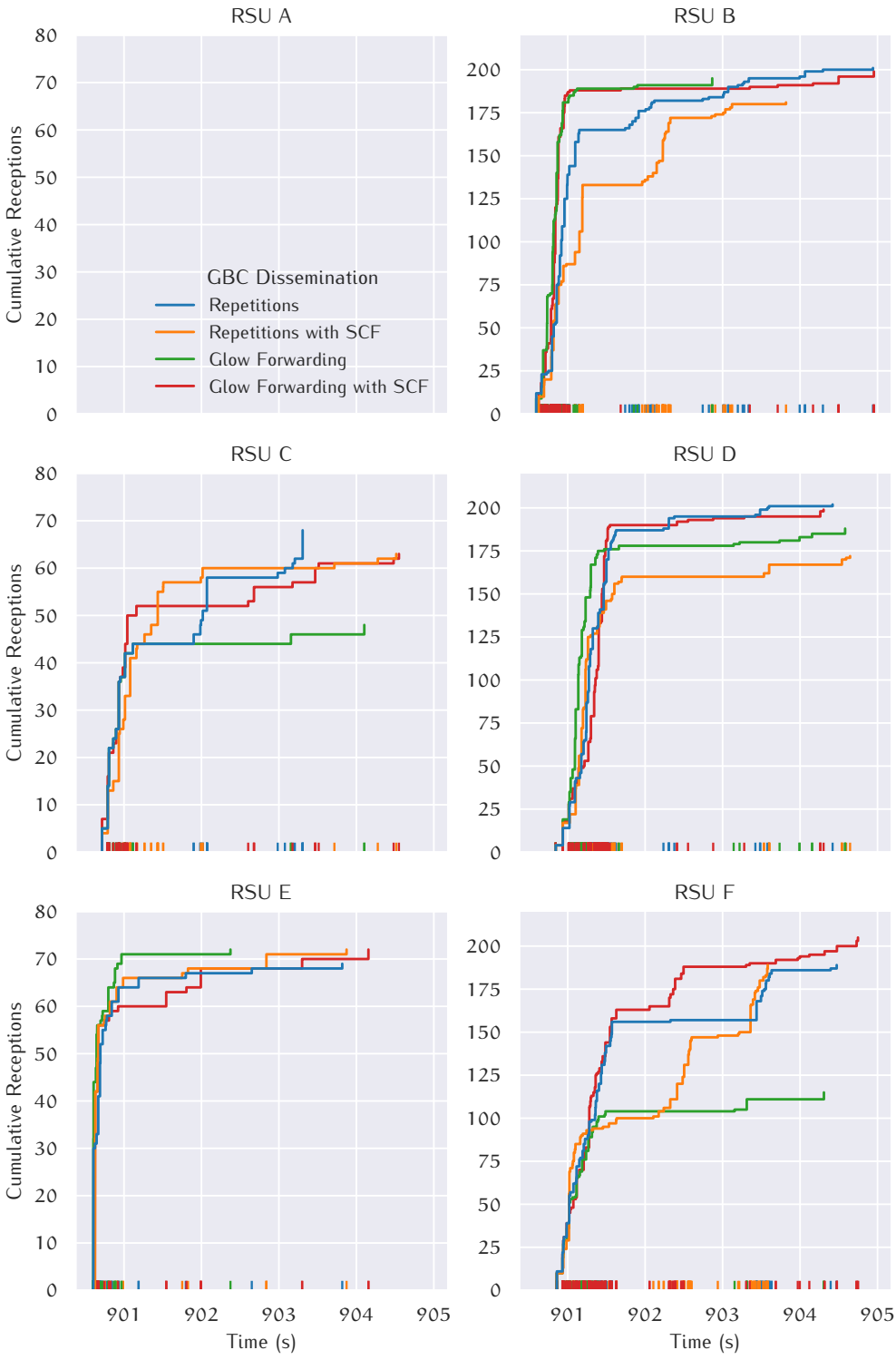


Figure 5.18: Receptions of RTCM messages over time with varying GBC dissemination features

5.3 URBAN ENVIRONMENT

The urban environment of the *LuST* scenario concentrates vehicles and communication. In total, 203 traffic lights are available on this map, which add substantial GBC traffic compared to the rural *DekiNet2*. Four RSUs collocated with traffic lights in the *LuST* scenario are also equipped with the GNSS augmentation service. In contrast to *DekiNet2*, the emphasis is on the GBC dissemination in the presence of obstructing buildings. Hence, RSUs near to multi-lane streets in the densely built-up city centre have been picked, as shown in Fig. 5.19. The size of the dissemination areas is kept identical to *DekiNet2*, but they are deliberately overlapping. Since more than one dissemination may be pending in the overlapping areas, this constellation allows to exercise forwarding's scalability. The following evaluations focus on the inner city area presented in Fig. 5.19, which still comprises 126 of the 203 traffic lights.

5.3.1 Starting Situation

Figure 5.20 shows the channel congestion right after the simulation's warmup period at 6am with 646 vehicles in total. Channel load is due to vehicles' CAMs and CPMs and infrastructure messages transmitted by traffic light RSUs. Two 'hotspots' with local CBRs around 35 % are reported in the areas at (6100 m, 5700 m) and (7000 m, 6900 m). From the global CBR perspective, the congestion looks more severe with congestion twice as high at some spots. The CBR information sharing via DCC_NET further creates clusters of similar congestion levels. During the warmup period, the LIMERIC algorithm initially allows comparatively long duty cycles, which leads to rather high congestion levels in this period. Global CBRs always lag behind the currently measured CBRs due to DCC_NET's distribution principle.

5.3.2 Impact of Rising Forwarding Load

The network load is increased by two DENM use cases starting from the described situation. First, 1 % of the vehicles start to report traction losses via non-emergency DENMs. Furthermore, ten vehicles are then abruptly stopped, so they are entitled to trigger emergency DENMs due to their hard braking. As long as they are decelerating by more than 7 m/s² and still faster than 20 km/h, an EEBL DENM will be disseminated every 100 ms. These DENMs are destined to all vehicles in a radius of 500 m and occur between simulation time 21 600.5 s and 21 601.5 s. In total, 58 emergency messages are generated in this setup by the first ten vehicles fulfilling the EEBL triggering conditions. On purpose, only vehicles in the map's centre are considered as

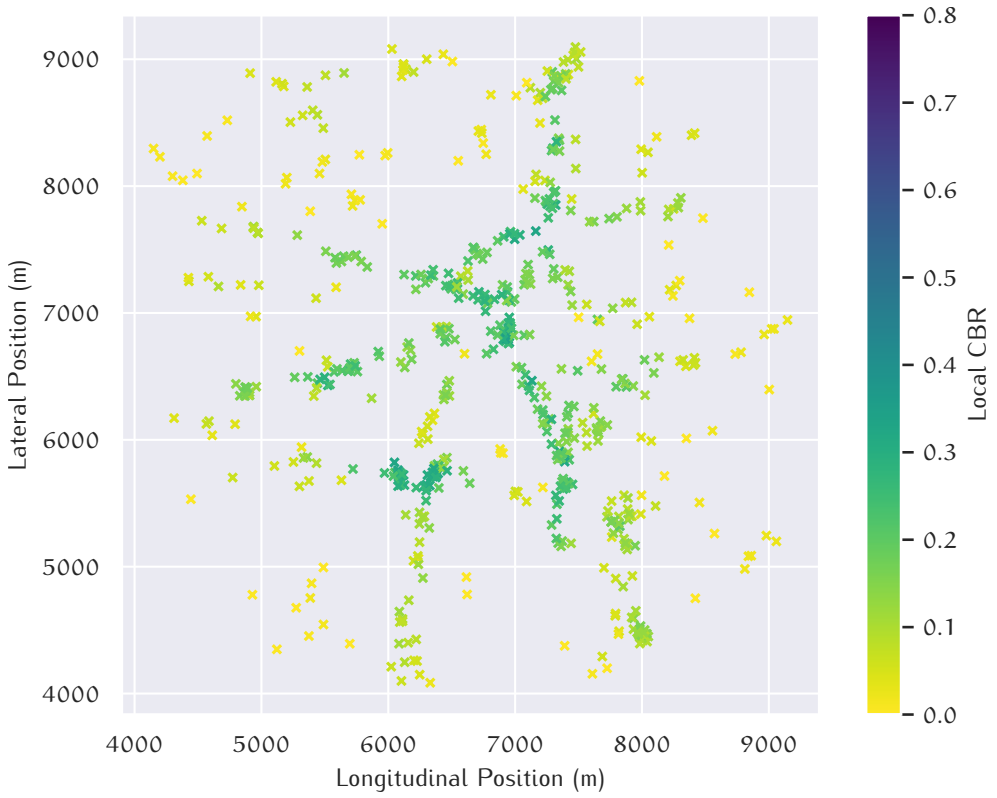
Figure 5.19: Four selected RSUs collocated with traffic lights on the LuST map

candidates to specifically study the EEBL dissemination in an environment with preloaded congestion. The ten EEBL transmitters are further on labelled from *V-0* to *V-9*.

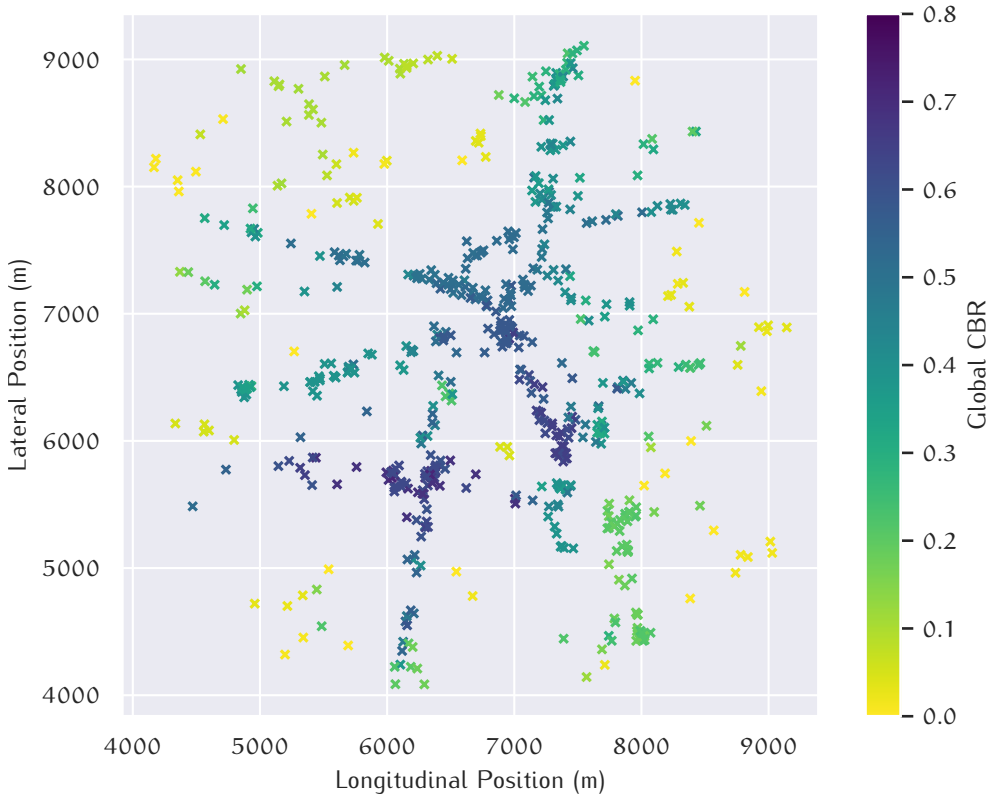
Figures 5.21 to 5.23 are snapshots related to the forwarding budget during the EEBL dissemination phase. The presented data belongs to forwarding budget data reported in the 100 ms window after 21 601 s, *i.e.* in the middle of the EEBL window.

Forwarding budgets, *i.e.* the allowed duty cycles for forwarding transmissions, range from 36 % to 100 % with $N_{fb} = 50$. Whenever a station's duty cycle is large enough, in this case $\delta >= 0.02$, the forwarding is unrestricted. However, in the map's middle, the allowed occupancy by forwarding transmissions is already restricted.

While Fig. 5.21 shows the permitted upper limit of channel occupancy by forwarding transmissions, Fig. 5.22 shows the actual channel occupancy due to those transmissions as monitored by each station. Unsurprisingly, the area with the EEBL sources *V-0* to *V-9* in the middle is occupied most. Up to 30 % of the channel's capacity is then used for packet forwarding. Yellow-greenish markers in the outskirts are due to other less frequent DENMs, SPATEMs and MAPEMs.



(a) Local CBRs



(b) Global CBRs

Figure 5.20: Channel congestion in LuST's centre at 6am with Adaptive DCC

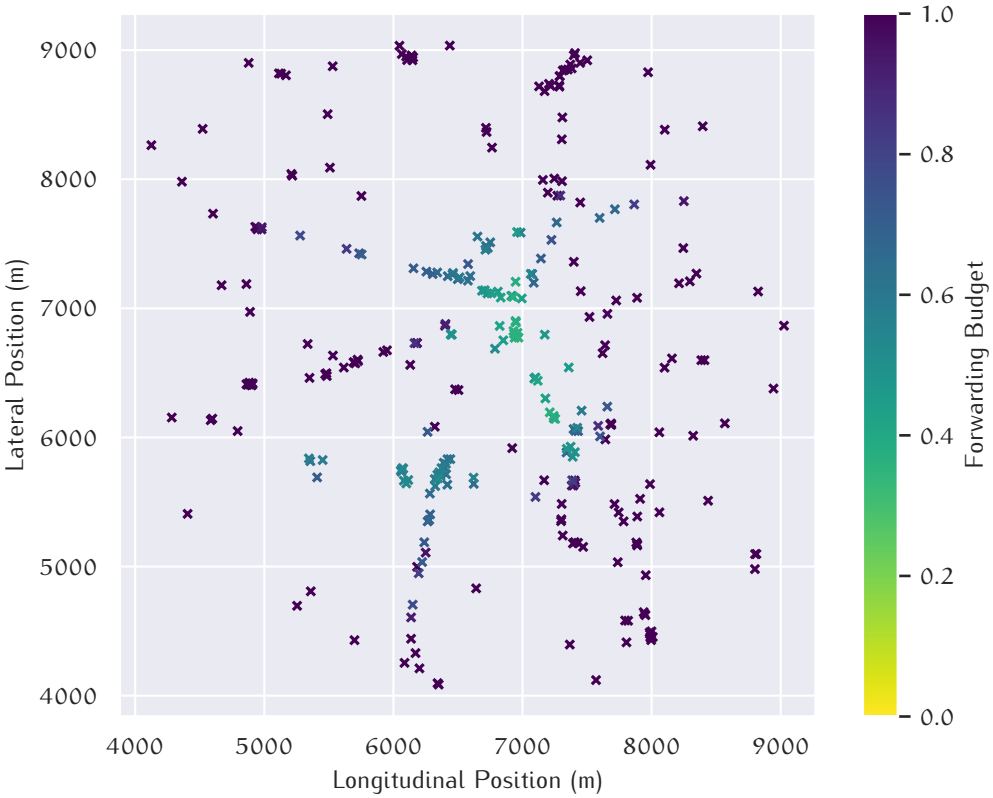


Figure 5.21: Forwarding budget with Fallco $N_{fb} = 50$ during EEBL dissemination

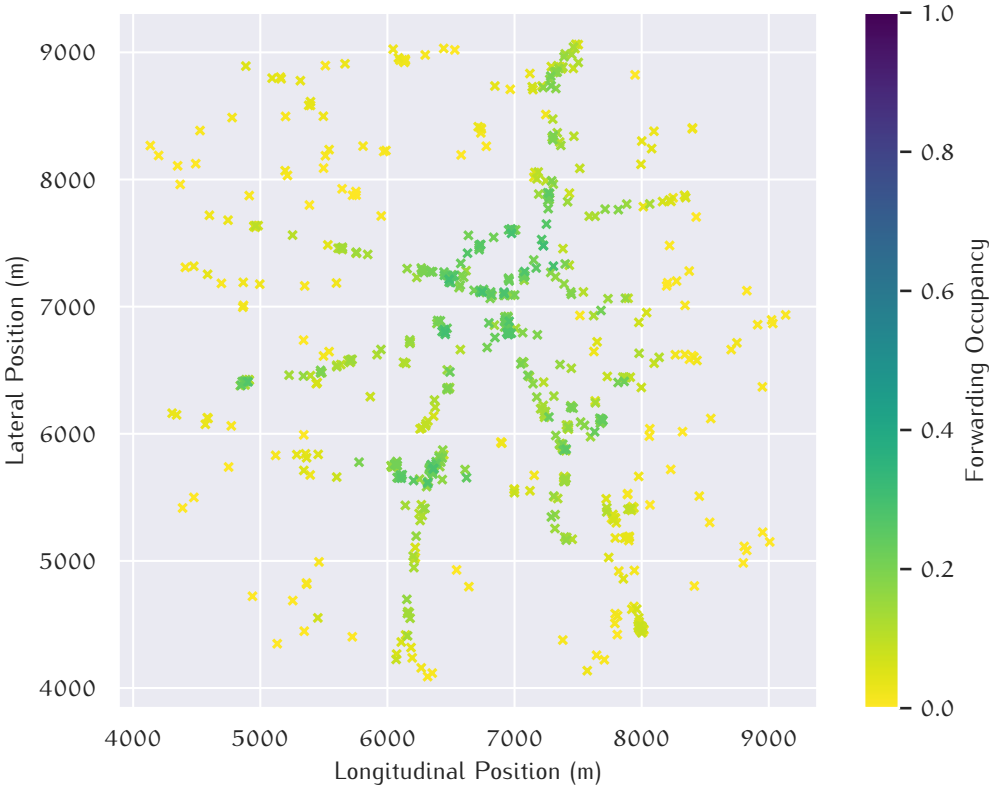


Figure 5.22: Forwarding occupancy with Fallco $N_{fb} = 50$ during EEBL dissemination

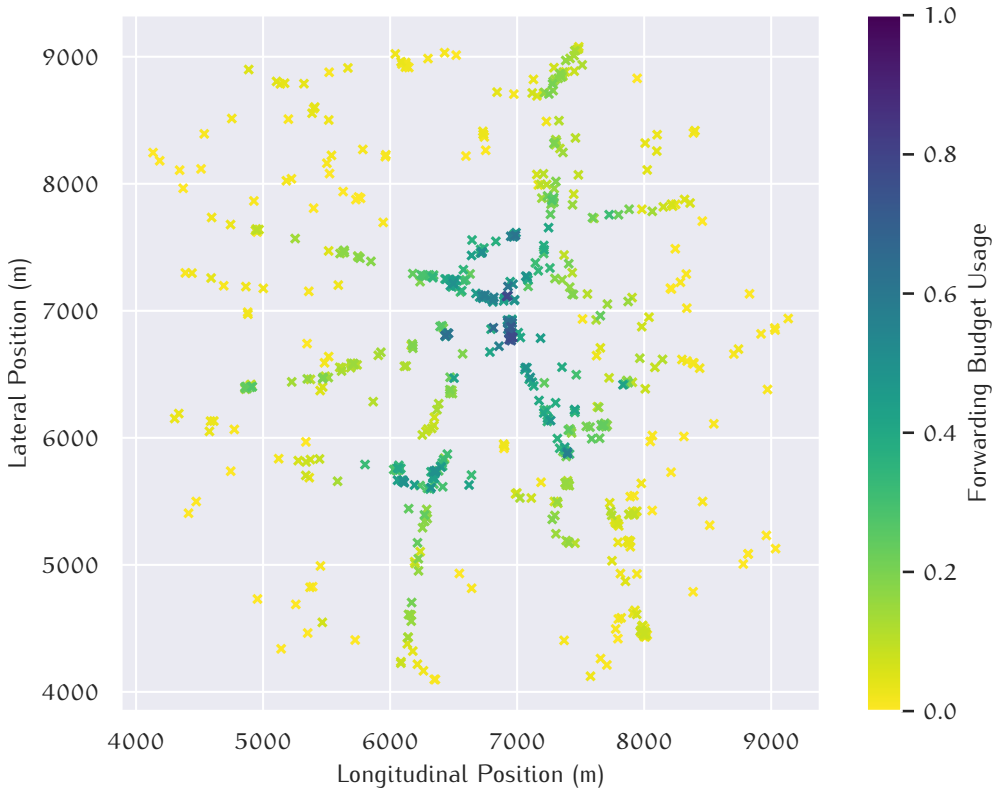


Figure 5.23: Forwarding usage with Fallco $N_{fb} = 50$ during EEBL dissemination

In addition to the previous two figures, Fig. 5.23 plots the forwarding budget usage, *i.e.* the forwarding occupancy relative to the determined budget. This budget usage is the input parameter to select the MCS for packet forwarding transmissions. The maximum usage level is at 78 % in the presented time window, and a tenth of all stations is above 45 %. Hence, data rates higher than 6 Mbit/s are employed, especially towards the map's centre. All forwarding transmission can still be handled, though.

5.3.3 Dissemination Performance of Emergency Messages

Emergency messages, which all belong to the EEBL use case in this setup, shall be disseminated quickly and as wide as possible to the addressed stations. Table 5.6 lists how many stations have been reached by a particular EEBL source. Depending on their braking action duration, they generate between one and ten DENMs due to this use case. For the four studied DCC variants, adaptive and Fallco with three budget configurations, the average number of distinct receivers by each DENM dissemination are given. Adaptive DCC further serves as a reference to calculate the relative gain of the Fallco DCC configurations.

The actual N_{fb} setting has only a minor impact on the number of reached stations: Only for *V-6*, the number of receivers grows with budget size. *V-5* is the only EEBL source, where the dissemination is significantly worse with

Table 5.6: Average number of receivers per generated EEBL and the gain of Fallco in relation to Adaptive DCC

Source	EEBLs \ DCC	Adaptive	Fallco $N_{fb} = 50$		Fallco $N_{fb} = 75$		Fallco $N_{fb} = 100$	
V-0	6	37.8	60.0	+58 %	60.0	+58 %	60.0	+58 %
V-1	9	61.9	90.0	+45 %	90.9	+47 %	90.2	+46 %
V-2	2	31.0	61.0	+97 %	61.0	+97 %	61.0	+97 %
V-3	1	31.0	65.0	+110 %	65.0	+110 %	65.0	+110 %
V-4	10	60.5	90.7	+50 %	90.9	+50 %	90.7	+50 %
V-5	8	58.6	75.4	+29 %	74.8	+28 %	66.8	+14 %
V-6	2	30.5	45.5	+49 %	51.5	+69 %	56.0	+84 %
V-7	7	66.1	92.3	+40 %	92.0	+39 %	92.1	+39 %
V-8	4	35.0	66.0	+89 %	65.8	+88 %	65.0	+86 %
V-9	9	71.9	93.1	+29 %	91.7	+28 %	93.2	+30 %

Table 5.7: EEBL dissemination summary for *V-0*

DCC	Receivers (mean)	Distance (max)	Latency (50 %/90 %/100 % quantiles)		
Adaptive	37.8	491.3 m	44.5 ms	557.9 ms	802.3 ms
Fallco $N_{fb} = 50$	60.0	497.9 m	94.1 ms	308.8 ms	537.4 ms
Fallco $N_{fb} = 75$	60.0	492.0 m	143.0 ms	356.6 ms	738.8 ms
Fallco $N_{fb} = 100$	60.0	494.3 m	132.2 ms	259.6 ms	586.2 ms

the highest budget compared to the lower budget alternatives, but still better than Adaptive DCC. Overall, Fallco DCC outperforms Adaptive DCC across the board.

In place of all ten sources, the latencies in the dissemination process of EEBLs by *V-0* are presented in the following. As listed in Table 5.6, *V-0* generates six EEBL messages in total. At the time of its first EEBL transmission, this vehicle is located at (7043 m, 6791 m), *i.e.* in a region of high forwarding budget usage according to Fig. 5.23.

Table 5.7 summarises the EEBL disseminations initiated by *V-0* in terms of maximum distance and latency quantiles. Practically, vehicles near the border of the destination area with 500 m radius can be reached with all four DCC variants. Complementary to the summary of Table 5.7, Fig. 5.24 details the dissemination process of *V-0* for each of its six DENMs as distance-latency plots: The distance between transmitter (*V-0*) and receiver is plotted on the abscissa, the ordinate shows the latency of the dissemination. Each dot corresponds to one unique receiver, and every receiver is only represented once by its first reception time point, *i.e.* its minimal latency. Those dots located on a common, horizontal line belong to receptions of a single transmission.

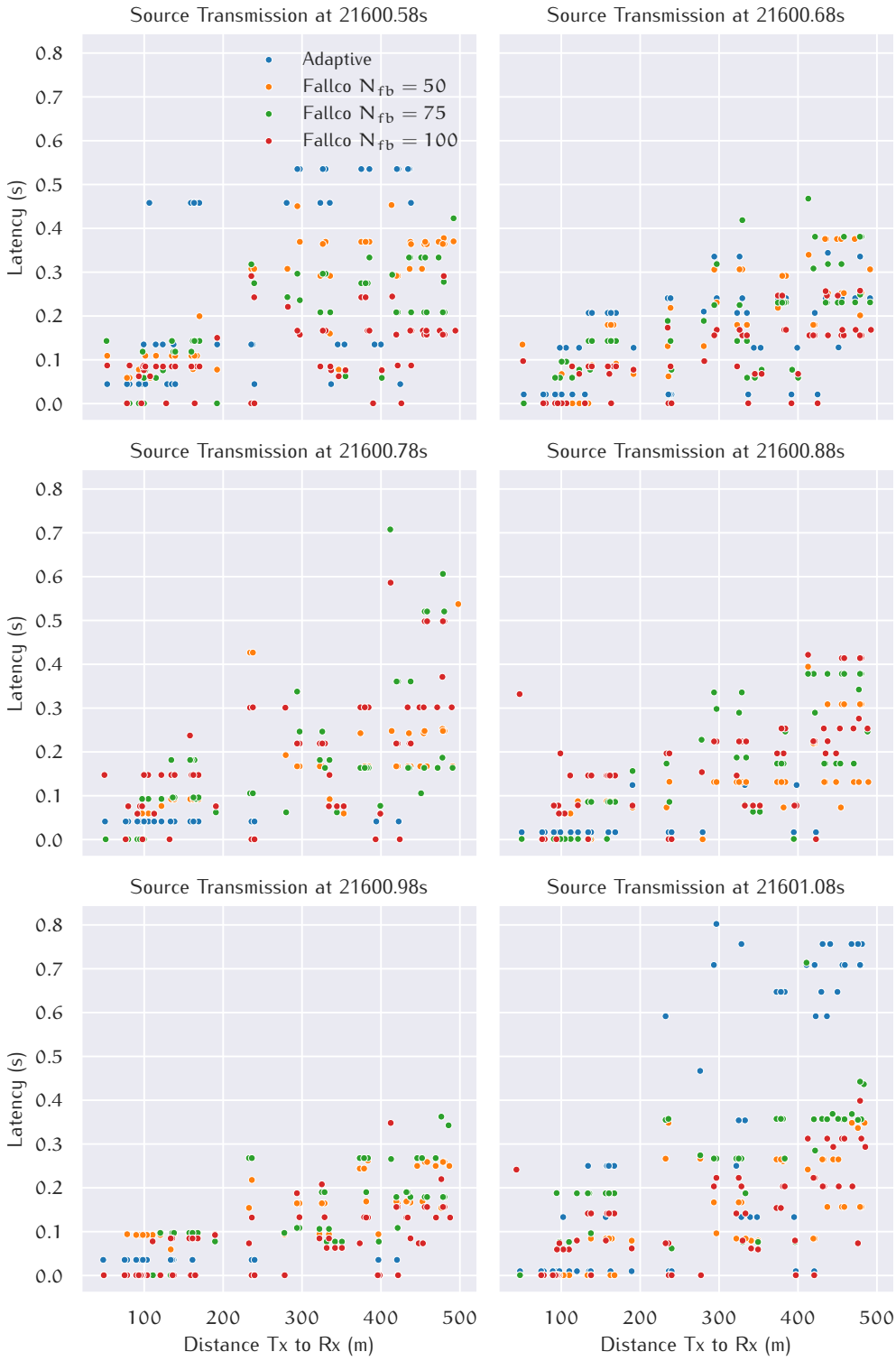


Figure 5.24: Distance-latency plots of EEBL messages by V-o

Adaptive DCC is always accompanied by an initial delay between the DENM generation and the actual first transmission step. Even emergency messages have to await the 25 ms interval since the preceding transmission. Fallco allows immediate transmission of emergency messages as long as the channel occupancy does not considerably exceed the permitted duty cycle. At this point, Fallco offers an advantage for emergency use cases over conventional DCC variants.

Though the forwarding budget of all three Fallco DCC variants is sufficient to cover the forwarding transmission demand, the dissemination is not the identical: Larger budgets allow for more forwarding transmissions at lower data rates, which are more robust. Those more robust MCSs give Fallco with $N_{fb} = 100$ only a slight advantage, though. In other words, switching MCSs to stretch the budget for more transmissions does not deteriorate the dissemination performance. The CBF forwarding scheme mitigates the risk of failing packet decoding if it can operate as designed.

Adaptive DCC, which has no forwarding budget, enqueues forwarding packets at DCC_ACC; only if no packet in the DP0, DP1 and DP2 queues is pending, then those forwarding packets are transmitted. Despite the ignoring of the CBF timing, the queue sojourn time increases the forwarding latency noticeable. Especially the first and the last DENM by *V-0* are affected by this unfavourable behaviour.

5.3.4 RTCM Coverage

Message dissemination over a large area, as it is exercised with the RTCM services, remains challenging in LuST's urban environment. Despite a more dense network topology than in the rural environment, which is favourable for routing, buildings' signal blockage seems to prevent high coverage ratios. In the best case, coverage ratios go up to around 40 %, as shown in Fig. 5.25.

RTCM coverage with Adaptive DCC is disappointing to a large extent. Except for RSU A, no coverage is achieved at all. In this case, the low-priority source traffic competes with forwarding traffic for a slot in the same DCC queue. RSU A is the most western RSU, *i.e.* RSUs B, C and D are located closer to the EEBL sources. Thus, the steady arrival of further forwarding packets at Adaptive DCC's DP3 queue displaces the RTCM transmissions. Fallco DCC employs its DCC queues only for source traffic, *i.e.* packets cannot be driven out of their queues by packet forwarding even if they share the same DP.

In sum, Adaptive DCC penalises the RTCM service unwarrantedly, which yields precedence to the SPATEMs and MAPEMs by choosing a lower DP. The separate handling of source and forwarding traffic by Fallco DCC pays off once more.

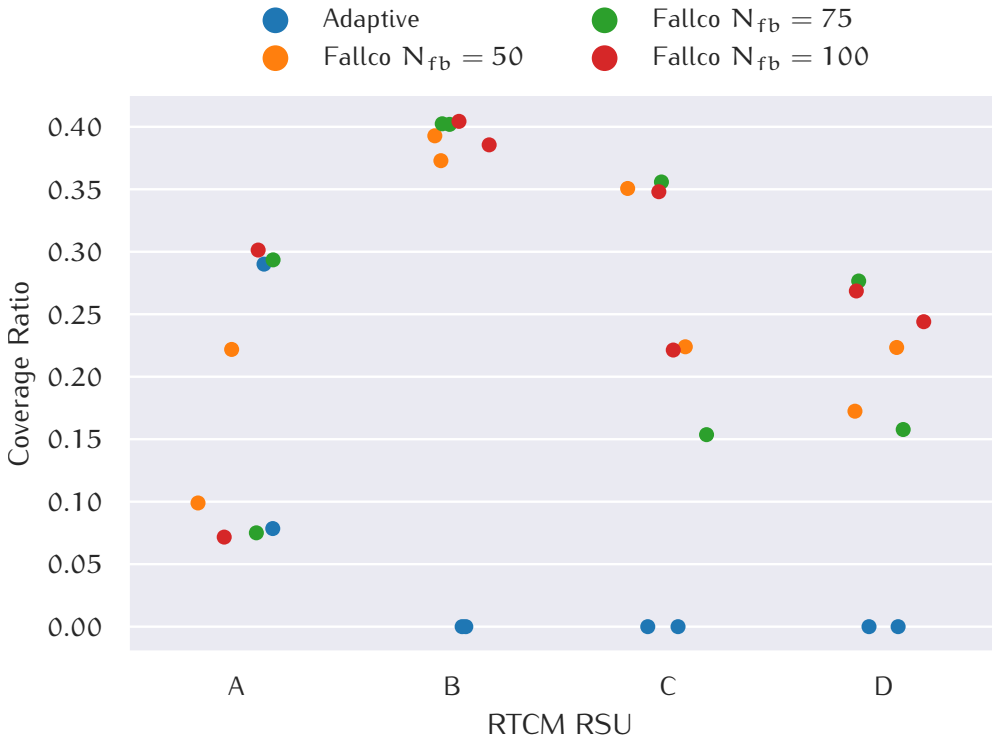


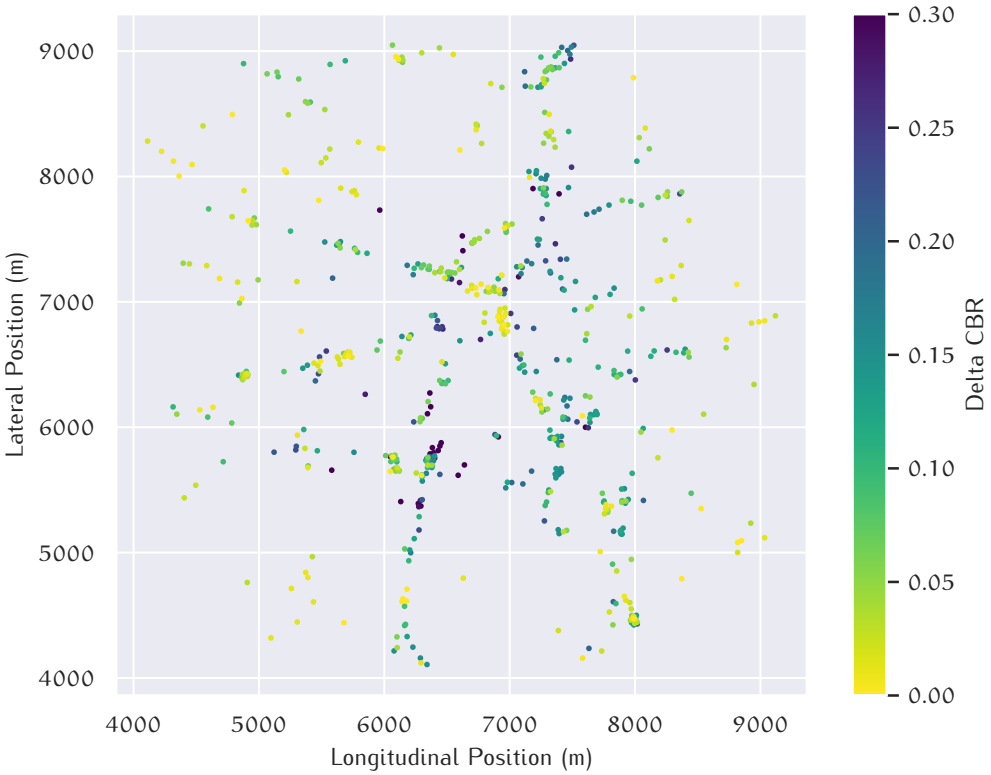
Figure 5.25: RTCM coverage in LuST scenario

5.3.5 Directional Disparity of CBRs

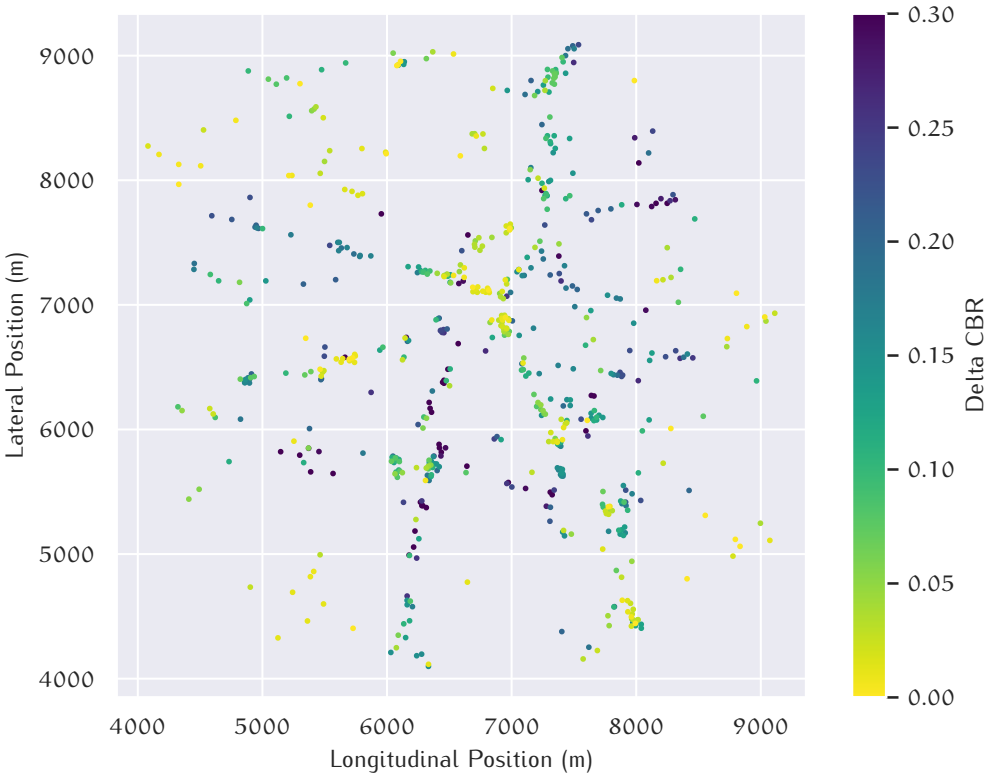
Section 4.2.1 introduced DCC-MAP as an alternative to DCC-MCO for sharing CBRs across the network. DCC-MAP is capable to replicate global CBRs similar to those produced by DCC-MCO, as demonstrated by Fig. 4.13 on Page 157. However, more sophisticated CBR metrics are possible with DCC-MAP, which make use of the position data stored along with received CBRs in LocTs. As a specific example, *Ahead* and *Behind* CBRs have been proposed in Section 4.2.1, which are based on measurements in cones ahead and behind the local vehicle, respectively. *Ahead* and *Behind* consider only near measurements, *i.e.* CBRs originating from other stations up to 500 m away. Alternatively, *Far Ahead* and *Far Behind* take measurements into account up to a distance of 1000 m.

The markers plotted in Fig. 5.26 indicate the difference between the channel load ahead and behind the respective vehicle at the end of the simulation run. For these visualisations, the delta CBR is calculated as $|\mathrm{CBR}_{\mathrm{Ahead}} - \mathrm{CBR}_{\mathrm{Behind}}|$ for the 500 m range and $|\mathrm{CBR}_{\mathrm{FarAhead}} - \mathrm{CBR}_{\mathrm{FarBehind}}|$ for the 1000 m range.

The yellow markers represent a close agreement of the channel congestion determined via DCC-MAP ahead and behind the respective vehicle's position. More interesting are the dark markers, because they highlight a large disparity of CBRs reported from ahead and behind. In these situations, the vehicles could still be allowed to transmit more packets in the less congested



(a) Delta CBR in near-range cones (*Ahead* and *Behind*)



(b) Delta CBR in far-range cones (*Far Ahead* and *Far Behind*)

Figure 5.26: Differences between ahead and behind CBRs aggregated via DCC-MAP

direction, provided it can control a transmission's direction, *e.g.* if multiple antennas are individually selectable. If directional communication is no option, the distance of reported CBRs could provide valuable input to control the transmission power. For example, transmissions with reduced power could be legitimate if CBRs only beyond a defined distance are high but not in the proximity.

However, embracing zones with lower channel congestion for additional communication is not part of the current ITS-G5 architecture. Neither do routing algorithms incorporate congestion in their decisions, nor has GN access to a standardised interface to control the direction of propagation of its packets. Nevertheless, the presented evaluation has shown that dormant potential in terms of non-uniform channel congestion exists and that it can be detected with DCC-MAP.

6 CONCLUSION

IVC has seen ups and downs over the years following the hype cycle as many innovative technologies do. Finally, the deployment of VANET communication progresses in Europe: Road infrastructure is getting upgraded by the C-Roads initiative[1]. In this context, the Austrian road operator ASFINAG sets up 525 RSUs compatible with ITS-G5 [191]. Furthermore, the year 2020 brought first mass production cars equipped with ITS-G5 technology to the market in Europe [203]. One can expect that the ITS-G5 market penetration will further ramp up in the foreseeable future. Especially as the density of communicating vehicles grows, the findings from this thesis become increasingly relevant.

6.1 FINDINGS AND CONTRIBUTIONS

As the review of standards and papers on ITS-G5 revealed, **QoS requirements** are only vaguely stated if at all. The first generation of ADASs using IVC is designed as advisory systems that do not require stringent QoS. VANETs building upon the ITS-G5 protocols depend highly on **keeping the channel congestion bounded**. Otherwise, common QoS criteria like latency and coverage deteriorate under excessive network load. The current ITS-G5 specifications, especially concerning DCC and GN, leave room for **improvements to make efficient use of channel resources**.

Several enhancements for ITS-G5 have been outlined, prototyped and demonstrated in this work: **DCC-MAP** enables sensing the network load by localised CBRs while preserving the size of the conventional DCC-MCO header. **Fallco DCC** incorporates the novelty of managing source and forwarding traffic individually, avoiding unwanted mutual interference. Furthermore, **GN flaws have been eliminated**: Redundant CBF packets are now tracked reliably, and thus broadcast storms are avoided. Also, LocTs do no longer contain outdated neighbours over long periods. Last but not least, **Glow Forwarding** and **revised SCF**, which takes the network topology into account, upgrade the dissemination of delay-tolerant packets.

Amalgamated to an enhanced ITS-G5 system, the measures proposed by this thesis have given proof of their positive advances regarding QoS. Confirmed by evaluations in large-scale scenarios with hundreds of vehicles,

1 https://www.c-roads.eu, accessed on 30th November 2020

Fallco DCC significantly cuts down the latency of CAMs and CPMs by several hundredths of a second. Even more, the worst-case latencies drop down to a few milliseconds while state-of-the-art systems quickly suffer from 30 times as high latencies. The introduction of **distinct forwarding budgets supports the timely dissemination of multi-hop packets**: More stations receive these packets in a shorter time. For example, about 30 % or in absolute numbers 800, more vehicles are reached in the rural DekiNet2 scenario compared to the closest, standardised competitor. If SCF checks the network topology before resuming packet forwarding, delay-tolerant packets can be disseminated to a **wider audience within the same time**. In combination with Glow Forwarding, the **coverage is above conventional repetition patterns**.

A substantial simulation environment, which goes by the name of Artery, has been developed and published to enable the methodical evaluation of ITS-G5's performance. Every layer found in ITS-G5 is considered by Artery, which makes it probably the **most comprehensive tool to simulate ITS-G5 VANETs** as of today. The simulation models of Artery cover the ITS-G5 entities and service of both, the standardised state-of-the-art as well as the proposed enhancements. Furthermore, Artery is not only used to evaluate QoS in the context of this thesis but has already impacted third parties to this day. Notably, Artery has been employed by others in their PhD theses [126, 130] and for research papers [158, 184, 181]. Commercial usage has also been confirmed [159]. Apart from ITS-G5, Wendland, Schaefer and Thomä used Artery to evaluate their DCC counterpart employed in LTE-V2X communications [185].

With all these findings and contributions, one can summarise that the objectives listed at the beginning of this thesis in Section 1.2 are achieved:

1. The state of the art with respect to QoS in ITS-G5 communications has been thoroughly laid out.

2. Several enhancements have been proposed to address the identified shortcomings related to channel usage and message dissemination performance.

3. Development of Artery and Vanetza has led to a comprehensive ITS-G5 simulation environment.

4. Simulation studies concerning proposed enhancements versus the standard baseline setup have shown the advantageous performance of the former.

Despite these encouraging advances, the development of ITS-G5 will not stop here. Nevertheless, the outcomes of this thesis can hopefully support the future directions of VANET research and deployment.

6.2 FUTURE WORK

RESPONSIVENESS AND OVERLOAD HANDLING The generation of a few GBC packets can induce a substantial amount of subsequent forwarding transmissions. LIMERIC has been originally designed to adapt the rate of periodic messages. In moderately crowded scenarios, as the evaluated rural and urban environments, LIMERIC's determined duty cycle is suitable to adjust the forwarding budget. However, this approach reaches its limits in extreme situations: Employing high multiples of LIMERIC's duty cycle as forwarding budget may overload the channel if usage of this budget rises suddenly. In turn, low multiples do not provide sufficient capacity for the packet forwarding demand. Future revisions of Fallco may incorporate the actual prevailing ratio of source to forwarding traffic and determine individual duty cycles for both traffic types.

As it stands now, the forwarding budget squeezes the airtime of forwarding transmissions by switching to higher rate MCSs. If the budget is finally exhausted, further forwarding packets are abruptly discarded. A more graceful forwarding behaviour may be achieved by gradually increasing the drop probability even before the budget runs short. Such a drop probability may even get coupled with TC of the forwarded packet, *e.g.* that emergency packets are more likely forwarded in extreme situations.

BREAKING UP WITH OMNIDIRECTIONALITY The presented concept of directional CBR revealed that the actual channel congestion is unevenly distributed around ITS stations. A gap of more than 20 percentage points, almost a third of target CBR level, is not uncommon. Unfortunately, the exclusively omnidirectional communication patterns of current station designs do not allow to make use of those idle channel resources. If vehicles are equipped with multiple antennas, *e.g.* motorcycles to overcome the shadowing of radio waves by the motor block, these are usually operated in antenna diversity mode. Finer control of the antenna used for transmission in combination with an upgraded GN protocol could employ idle resources selectively, though. Thus, directional CBR reports can become a building block for next-generation ITS-G5 stations, which exploit the frequency spectrum spatially.

TRANSFER TO STANDARDISATION The transfer of the presented and evaluated enhancements to standardisation is another next step. Actually required efforts for these transfers likely depend on how far the necessary modifications reach. Fixing unmistaken bugs such as CBF broadcast storms and never-vanishing neighbour entries in LocTs is more urgent than the integration of new features. However, features such as Glow Forwarding, handling of SCF and cancellation of pending packets have been designed with back-

ward compatibility to the current GN protocol revision in mind. Hopefully, this careful design approach pays off by easing acceptance for these in the standardisation process. Even if GN features proposed in this thesis may not become straight part of the GN specification, they may at least inspire the evolution of GN. In the future, the distinction between SCF, CBF, Glow Forwarding should vanish in favour of a unified GN, which requires less parameterisation. With the proposed GN enhancements, the interface exposed to applications has become even more complex, unfortunately. Ideally, applications should state their dissemination intents and constraints instead of low-level GN parameters.

The DCC cross-layer has seen even more drastic changes by this thesis than GN. Though the Fallco DCC entity incorporates the LIMERIC algorithm, taken as a whole, its design is different to its alternatives – including reactive and adaptive DCC from standardisation. Some parts, such as the scheduling of periodic services, are comparatively easy to migrate because it only affects a station's internals. Extensions of the internal behaviour, *e.g.* assignment of individual scheduling weights to each service, can also be realised without much effort in the future. Other parts, with the concept of forwarding budget leading the way, rely on the collaboration of stations. Since the benefit of such a forwarding budget could be clearly demonstrated, this thesis provides good arguments to relax the mandatory inter-packet gap of 25 ms demanded by ETSI. Nevertheless, before actual deployments of this concept in the field, *EN 302 571* must be revised accordingly.

A PUBLISHED WORK

[1] Raphael Riebl and Christian Facchi. 'Implementation of Day One ITS-G5 Systems for Testing Purposes'. In: *Proceedings of the 2nd GI/ITG KuVS Fachgespräch Inter-Vehicle Communication (FG-IVC 2014)*. Feb. 2014, pp. 33–36. ISBN: 978-2-87971-124-9.

[2] Raphael Riebl and Christian Facchi. 'Regain Control of Growing Dependencies in OMNeT++ Simulations'. In: *Proceedings of the 2nd OMNeT++ Community Summit*. Sept. 2015.

[3] Raphael Riebl, Hendrik-Jörn Günther, Christian Facchi and Lars Wolf. 'Artery: Extending Veins for VANET applications'. In: *International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS 2015)*. June 2015, pp. 450–456. DOI: 10.1109/MTITS.2015.7223293. URL: https://github.com/riebl/artery.

[4] Raphael Riebl, Stefan Neumeier and Christian Facchi. 'Inter-Vehicle Communication on the Run - Experiences From Tweaking Veins Runtime Performance'. In: *Proceedings of the 3rd GI/ITG KuVS Fachgespräch Inter-Vehicle Communication (FG-IVC 2015)*. Ed. by Raphael Frank, Christoph Sommer, Frank Kargl, Stefan Dietzel and Rens W. van der Heijden. 2015.

[5] Hendrik-Jörn Günther, Björn Mennenga, Oliver Trauer, Raphael Riebl and Lars Wolf. 'Realizing Collective Perception in a Vehicle'. In: *IEEE Vehicular Networking Conference (VNC)*. Dec. 2016. DOI: 10.1109/VNC.2016.7835930.

[6] Hendrik-Jörn Günther, Raphael Riebl, Lars Wolf and Christian Facchi. 'Collective Perception and Decentralized Congestion Control in Vehicular Ad-hoc Networks'. In: *IEEE Vehicular Networking Conference (VNC)*. Dec. 2016. DOI: 10.1109/VNC.2016.7835931.

[7] Raphael Riebl, Markus Monz, Simon Varga, Leandros Maglaras, Helge Janicke, Ali H. Al-Bayatti and Christian Facchi. 'Improved Security Performance for VANET Simulations'. In: *Proceedings on the 4th IFAC Symposium on Telematics Applications*. Nov. 2016, pp. 233–238. DOI: 10.1016/j.ifacol.2016.11.173.

[8] Thomas Speth, Raphael Riebl, Thomas Brandmeier, Christian Facchi, Ali H. Al-Bayatti and Ulrich Jumar. 'Enhanced Inter-Vehicular Relative Positioning'. In: *19th International Conference on Intelligent Transportation Systems (ITSC)*. Nov. 2016, pp. 867–872. DOI: 10.1109/ITSC.2016.7795657.

[9] Thomas Speth, Raphael Riebl, Christian Facchi, Ulrich Jumar and Ali H. Al-Bayatti. 'VANET Coverage Analysis for GPS Augmentation Data in Rural Area'. In: *Proceedings on the 4th IFAC Symposium on Telematics Applications*. Nov. 2016, pp. 245–250. DOI: 10.1016/j.ifacol.2016.11.112.

[10] Hendrik-Jörn Günther, Julian Timpner, Martin Wegner, Raphael Riebl and Lars Wolf. 'Extending a holistic microscopic IVC simulation environment with local perception sensors and LTE capabilities'. In: *Vehicular Communications* 9 (2017), pp. 211–221. ISSN: 2214-2096. DOI: 10.1016/j.vehcom.2017.01.003.

[11] Christina Obermaier, Raphael Riebl and Christian Facchi. 'Dynamic Scenario Control for VANET Simulations'. In: *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. June 2017, pp. 681–686. DOI: 10.1109/MTITS.2017.8005599.

[12] Raphael Riebl, Christina Obermaier, Stefan Neumeier and Christian Facchi. 'Vanetza: Boosting Research on Inter-Vehicle Communication'. In: *Proceedings of the 5th GI/ITG KuVS Fachgespräch Inter-Vehicle Communication (FG-IVC 2017)*. Ed. by Anatoli Djanatliev, Kai-Steffen Hielscher, Christoph Sommer, David Eckhoff and Reinhard German. Apr. 2017, pp. 37–40.

[13] Hendrik-Jörn Günther, Raphael Riebl, Lars Wolf and Christian Facchi. 'The Effect of Decentralized Congestion Control on Collective Perception in Dense Traffic Scenarios'. In: *Computer Communications* 122 (2018), pp. 76–83. ISSN: 0140-3664. DOI: 10.1016/j.comcom.2018.03.009.

[14] Christina Obermaier, Raphael Riebl and Christian Facchi. 'Fully Reactive Hardware-in-the-Loop Simulation for VANET Devices'. In: *21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 3755–3760. DOI: 10.1109/ITSC.2018.8569663.

[15] Raphael Riebl, Giovanni Nardini and Antonio Virdis. 'Simulating LTE-Enabled Vehicular Communications'. In: *Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem*. Ed. by Antonio Virdis and Michael Kirsche. Cham: Springer International Publishing, 2019, pp. 407–423. ISBN: 978-3-030-12842-5. DOI: 10.1007/978-3-030-12842-5_13.

[16] Raphael Riebl, Christina Obermaier and Hendrik-Jörn Günther. 'Artery: Large Scale Simulation Environment for ITS Applications'. In: *Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem*. Ed. by Antonio Virdis and Michael Kirsche. Cham: Springer International Publishing, 2019, pp. 365–406. ISBN: 978-3-030-12842-5. DOI: 10.1007/978-3-030-12842-5_12.

REFERENCES

[17] Minoru Nakagami. 'The m-Distribution - A General Formula of Intensity Distribution of Rapid Fading'. In: *Statistical Methods in Radio Wave Propagation*. Ed. by W.C. Hofmann. Pergamon, 1960, pp. 3–36. ISBN: 978-0-08-009306-2. DOI: 10.1016/B978-0-08-009306-2.50005-4.

[18] Errol L. Lloyd and S. Ramanathan. 'On the Complexity of Distance-2 Coloring'. In: *Fourth International Conference on Computing and Information*. May 1992, pp. 71–74. DOI: 10.1109/ICCI.1992.227702.

[19] Sally Floyd and Van Jacobson. 'Random Early Detection for Congestion Avoidance'. In: 1.4 (Aug. 1993), pp. 397–413. DOI: 10.1109/90.251892.

[20] ISO/IEC JTC 1. *Internatioal Standard ISO/IEC 7498-1 - Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model*. 2nd ed. ISO/IEC. Nov. 1994.

[21] Robert Braden, David Clark and Scott Shenker. *Integrated Services in the Internet Architecture: an Overview*. RFC 1633. June 1994. DOI: 10.17487/RFC1633.

[22] Chunhung Richard Lin and Mario Gerla. 'Asynchronous Multimedia Multihop Wireless Networks'. In: *Proceedings of INFOCOM '97*. Apr. 1997. DOI: 10.1109/INFCOM.1997.635121.

[23] Steven Blake, David L. Black, Mark A. Carlson, Elwyn Davies, Zheng Wang and Walter Weiss. *An Architecture for Differentiated Services*. RFC 2475. Dec. 1998. DOI: 10.17487/RFC2475.

[24] Xiaopeng Ma and Errol L. Lloyd. 'An incremental algorithm for broadcast scheduling in packet radio networks'. In: *IEEE Military Communications Conference*. Oct. 1998. DOI: 10.1109/MILCOM.1998.722572.

[25] Federal Communications Commission. *FCC 99-305: Amendment of Part 2 and 90 of the Commission's Rules to Allocate the 5.850-5.925 GHz Band to the Mobile Service for Dedicated Short Range Communications of Intelligent Transport Services*. Oct. 1999.

[26] João L. Sobrinho and A. S. Krishnakumar. 'Quality-of-Service in Ad Hoc Carrier Sense Multiple Access Wireless Networks'. In: *IEEE Journal on Selected Areas in Communications* 17.8 (Aug. 1999), pp. 1353–1368. DOI: 10.1109/49.779919.

[27] Seoung-Bum Lee, Gahng-Seop Ahn, Xiaowei Zhang and Andrew T. Campbell. 'INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks'. In: *Journal of Parallel and Distributed Computing* 60.4 (2000), pp. 374–406. DOI: 10.1006/jpdc.1999.1613.

[28] M.H. MacGregor and W. Shi. 'Deficits for bursty latency-critical flows: DRR++'. In: *IEEE International Conferenece on Networks 2000 (ICON 2000). Networking Trends and Challenges in the New Millenium*. Sept. 2000, pp. 287–293. DOI: 10.1109/ICON.2000.875803.

[29] Theodore S. Rappaport. *Wireless Communications - Principles and Practice*. 2nd ed. Prentice Hall Communications Engineering and Emerging Technologies Series. Upper Saddle River, NJ, USA: Prentice Hall, 2002. ISBN: 0-13-042232-0.

[30] Christian Maihöfer, Christian Cseh, Walter Franz and Reinhold Eberhardt. 'Performance Evaluation of Stored Geocast'. In: *IEEE Vehicular Technology Conference (VTC Fall)*. Oct. 2003, pp. 2901–2905. DOI: 10.1109/VETECF.2003.1286151.

[31] Leonard E. Miller. *Validation of 802.11a/UWB Coexistence Simulation*. Tech. rep. Oct. 2003. URL: http://web.archive.org/web/20041019043620/http://www.antd.nist.gov/wctg/manet/docs/coexvalid_031017.pdf (visited on 07/08/2019).

[32] Prasant Mohapatra, Jian Li and Chao Gui. 'QoS in Mobile Ad Hoc Networks'. In: *IEEE Wireless Communications* 10.3 (June 2003), pp. 44–52. DOI: 10.1109/MWC.2003.1209595.

[33] Mate Boban, João Barros and Ozan K. Tonguz. 'Geometry-Based Vehicle-to-Vehicle Channel Modeling for Large-Scale Simulation'. In: *IEEE Transactions on Vehicular Technology* 63.9 (Nov. 2004), pp. 4146–4164. DOI: 10.1109/TVT.2014.2317803.

[34] Andrzej Kochut, Arunchandar Vasan, A. Udaya Shankar and Ashok Agrawala. 'Sniffing out the correct physical layer capture model in 802.11b'. In: *Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP)*. Oct. 2004, pp. 252–261. DOI: 10.1109/ICNP.2004.1338115.

[35] Vikas Taliwal, Daniel Jiang, Heiko Mangold, Chi Chen and Raja Sengupta. 'Empirical Determination of Channel Characteristics for DSRC Vehicle-to-vehicle Communication'. In: *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*. VANET'04. Philadelphia, PA, USA: ACM, Oct. 2004, pp. 88–88. DOI: 10.1145/1023875.1023890.

[36] Marc Torrent-Moreno, Daniel Jiang and Hannes Hartenstein. 'Broadcast Reception Rates and Effects of Priority Access in 802.11-based Vehicular Ad-hoc Networks'. In: *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*. VANET '04. Philadelphia, PA, USA: ACM, 2004. DOI: 10.1145/1023875.1023878.

[37] Li Bin Jiang and Soung Chang Liew. 'Proportional fairness in wireless LANs and ad hoc networks'. In: *IEEE Wireless Communications and Networking Conference*. Vol. 3. Mar. 2005, pp. 1551–1556. DOI: 10.1109/WCNC.2005.1424745.

[38] Hans-J. Reumerman, Marco Roggero and Marco Ruffini. 'The Application-Based Clustering Concept and Requirements for Intervehicle Networks'. In: *IEEE Communications Magazine* 43.4 (2005), pp. 108–113. DOI: 10.1109/MCOM.2005.1421913.

[39] Emma Carlson, Christian Prehofer, Christian Bettstetter, Holger Karl and Adam Wolisz. 'A Distributed End-to-End Reservation Protocol for IEEE 802.11-Based Wireless Mesh Networks'. In: *IEEE Journal on Selected Areas in Communications* 24.11 (Nov. 2006), pp. 2018–2027. DOI: 10.1109/JSAC.2006.881633.

[40] Daniel Jiang, Vikas Taliwal, Andreas Meier, Wieland Holfelder and Ralf Hertwich. 'Design of 5.9 GHz DSRC-based Vehicular Safety Communication'. In: *IEEE Wireless Communications* 13.5 (2006), pp. 36–43. DOI: 10.1109/WC-M.2006.250356.

[41] Marc Torrent-Moreno, Paolo Santi and Hannes Hartenstein. 'Distributed Fair Transmit Power Adjustment for Vehicular Ad Hoc Networks'. In: *3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*. Sept. 2006, pp. 479–488. DOI: 10.1109/SAHCN.2006.288504.

[42] Qi Chen, Felix Schmidt-Eisenlohr, Daniel Jiang, Marc Torrent-Morreno, Luca Delgrossi and Hannes Hartenstein. 'Overhaul of IEEE 802.11 Modeling and Simulation in ns-2'. In: *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems (MSWiM)*. MSWiM '07. Chania, Crete Island, Greece: ACM, Oct. 2007, pp. 159–168. DOI: 10.1145/1298126.1298155.

[43] Lin Cheng, Benjamin E. Henty, Daniel D. Stancil, Fan Bai and Priyantha Mudalige. 'Mobile Vehicle-To-Vehicle Narrow-Band Channel Measurement and Characterization of the 5.9 GHz Dedicated Short Range Communication (DSRC) Frequency Band'. In: *IEEE Journal on Selected Areas in Communications* 25.8 (Oct. 2007), pp. 1501–1516. DOI: 10.1109/JSAC.2007.071002.

[44] John Evans and Clarence Filsfils. *Deploying IP and MPLS QOS for Multiservice Networks: Theory & Practice*. The Morgan Kaufmann Series in Networking. San Francicso, CA, USA: Morgan Kaufmann Publishers, 2007. ISBN: 0-12-370549-5.

[45] Marie Nestor Mariyasagayam, Tatsuaki Osafune and Massimiliano Lenardi. 'Enhanced Multi-Hop Vehicular Broadcast (MHVB) for Active Safety Applications'. In: *7th International Conference on ITS Telecommunications*. June 2007. DOI: 10.1109/ITST.2007.4295866.

[46] Marc Torrent Moreno. 'Inter-Vehicle Communications: Achieving Safety in a Distributed Wireless Environment. Challenges, Systems and Protocols'. PhD thesis. Karlsruhe: Universität Karlsruhe, Fakultät für Informatik, 2007. ISBN: 978-3-86644-175-0. DOI: 10.5445/KSP/1000007058.

[47] Yung Yi and Sanjay Shakkottai. 'Hop-by-Hop Congestion Control Over a Wireless Multi-Hop Network'. In: *IEEE/ACM Transactions on Networking* 15.1 (Feb. 2007), pp. 133–144. DOI: 10.1109/TNET.2006.890121.

[48] Sangtae Ha, Injong Rhee and Lisong Xu. 'CUBIC: A New TCP-Friendly High-Speed TCP Variant'. In: *SIGOPS Operating Systems Review* 42.5 (July 2008), pp. 64–74. DOI: 10.1145/14000097.1400105.

[49] Daniel Jiang, Qi Chen and Luca Delgrossi. 'Optimal Data Rate Selection for Vehicle Safety Communications'. In: *Proceedings of the Fifth ACM International Workshop on Vehicular Inter-Networking*. VANET '08. San Francisco, California, USA: Association for Computing Machinery, 2008, pp. 30–38. DOI: 10.1145/1410043.1410050.

[50] Jürgen Kunisch and Jörg Pamp. 'Wideband Car-to-Car Radio Channel Measurements and Model at 5.9 GHz'. In: *68th IEEE Vehicular Technology Conference*. Sept. 2008. DOI: 10.1109/VETECF.2008.64.

[51] XiPeng Xiao. *Technical, Commercial and Regulatory Challenges of QoS: An Internet Service Model Perspective*. The Morgan Kaufmann Series in Networking. Elsevier Science, 2008, p. 296. ISBN: 978-0-12-373693-2. DOI: 10.1016/B978-0-12-373693-2.X0001-8.

[52] Mark Allman, Vern Paxson and Ethan Blanton. *TCP Congestion Control*. RFC 5681. Sept. 2009. DOI: 10.17487/RFC5681.

[53] ETSI. *TR 102 638 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions*. Tech. rep. June 2009.

[54] Andreas Festag, Roberto Baldessari, Wenhui Zhang and Long Le. 'CAR-2-X Communication SDK - A Software Toolkit for Rapid Application Development and Experimentations'. In: *IEEE International Conference on Communications Workshops*. June 2009. DOI: 10.1109/ICCW.2009.5208070.

[55] Bahareh Sadeghi. 'Congestion Control in Wireless Mesh Networks'. In: *Guide to Wireless Mesh Networks*. Ed. by Sudip Misra, Subhas Chandra Misra and Isaac Woungang. London: Springer, 2009, pp. 277–298. ISBN: 978-1-84800-909-7. DOI: 10.1007/978-1-84800-909-7_11.

[56] Fan Bai, Daniel D. Stancil and Hariharan Krishnan. 'Toward Understanding Characteristics of Dedicated Short Range Communications (DSRC) From a Perspective of Vehicular Network Engineers'. In: *Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking*. MobiCom '10. Chicago, Illinois, USA: Association for Computing Machinery, Sept. 2010, pp. 329–340. DOI: 10.1145/1859995.1860033.

[57] ETSI. *EN 302 665 - Intelligent Transport Systems (ITS); Communications Architecture*. V1.1.1. European Telecommunications Standards Institute. Sept. 2010.

[58] Guangyu Pei and Thomas R. Henderson. *Validation of OFDM model in ns-3*. Tech. rep. 2010. URL: http://www.nsnam.org/~pei/80211ofdm.pdf (visited on 07/08/2019).

[59] Falko Dressler, Christoph Sommer, David Eckhoff and Ozan K. Tonguz. 'Towards Realistic Simulation of Intervehicle Communication'. In: *IEEE Vehicular Technology Magazine* 6.3 (Sept. 2011), pp. 43–51. DOI: 10.1109/MVT.2011.941898.

[60] ETSI. *EN 302 931 - Intelligent Transport Systems (ITS); Vehicular Communications; Geographical Area Definition*. V1.1.1. European Telecommunications Standards Institute. July 2011.

[61] ETSI. *TS 102 687 - Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part*. V1.1.1. European Telecommunications Standards Institute. July 2011.

[62] Jim Gettys and Kathleen Nichols. 'Bufferbloat: Dark Buffers in the Internet'. In: *Queue* 9.11 (Nov. 2011), pp. 40–54. ISSN: 1542-7730. DOI: 10.1145/2063166.2071893.

[63] John B. Kenney, Gaurav Bansal and Charles E. Rohrs. 'LIMERIC: A Linear Message Rate Control Algorithm for Vehicular DSRC Systems'. In: *Proceedings of the Eighth ACM International Workshop on Vehicular Inter-networking*. VANET '11. Las Vegas, Nevada, USA: ACM, 2011, pp. 21–30. ISBN: 978-1-4503-0869-4. DOI: 10.1145/2030698.2030702.

[64] Andreas Kwoczek, Zbyněk Raida, Jaroslav Láčík, Michal Pokorný, Jan Puskely and Petr Vágner. 'Influence of Car Panorama Glass Roofs on Car2Car Communication (Poster)'. In: *IEEE Vehicular Networking Conference (VNC)*. Nov. 2011, pp. 246–251. DOI: 10.1109/VNC.2011.6117107.

[65] Filipe Neves, André Cardote, Ricardo Moreira and Susana Sargento. 'Real-World Evaluation of IEEE 802.11p for Vehicular Networks'. In: *Proceedings of the 8th ACM International Workshop on Vehicular Inter-Networking*. VANET '11. Las Vegas, Nevada, USA: Association for Computing Machinery, Sept. 2011, pp. 89–90. DOI: 10.1145/2030698.2030717.

[66] Robert K. Schmidt, Achim Brakemeier, Tim Leinmüller, Frank Kargl and Günter Schäfer. 'Advanced Carrier Sensing to Resolve Local Channel Congestion'. In: *Proceedings of the Eighth ACM International Workshop on Vehicular Inter-Networking*. VANET '11. Las Vegas, Nevada, USA: Association for Computing Machinery, 2011, pp. 11–20. DOI: 10.1145/2030698.2030701.

[67] Katrin Sjöberg, Elisabeth Uhlemann and Erik G. Ström. 'How Severe is the Hidden Terminal Problem in VANETs when Using CSMA and STDMA?' In: *IEEE Vehicular Technology Conference (VTC Fall)*. Sept. 2011. DOI: 10.1109/VETECF.2011.6093256.

[68] Christoph Sommer, David Eckhoff, Reinhard German and Falko Dressler. 'A Computationally Inexpensive Empirical Model of IEEE 802.11p Radio Shadowing in Urban Environments'. In: *Eigth International Conference on Wireless On-Demand Network Systems and Services*. Jan. 2011, pp. 84–90. DOI: 10.1109/WONS.2011.5720204.

[69] Christoph Sommer, German Reinhard and Falko Dressler. 'Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis'. In: *IEEE Transactions on Mobile Computing* 10.1 (Jan. 2011), pp. 3–15. DOI: 10.1109/TMC.2010.133.

[70] Tessa Tielert, Daniel Jiang, Qi Chen, Luca Delgrossi and Hannes Hartenstein. 'Design Methodology and Evaluation of Rate Adaptation Based Congestion Control for Vehicle Safety Communications'. In: *IEEE Vehicular Networking Conference (VNC)*. Nov. 2011, pp. 116–123. DOI: 10.1109/VNC.2011.6117132.

[71] Mate Boban. 'Realistic and Efficient Channel Modeling for Vehicular Networks'. PhD thesis. Carnegie Mellon University, Dec. 2012.

[72] Hector Agustin Cozzetti, Claudia Campolo, Riccardo Scopigno and Antonella Molinaro. 'Urban VANETs and Hidden Terminals: Evaluation through a Realistic Urban Grid Propagation Model'. In: *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. July 2012, pp. 93–98.

[73] ETSI. *TS 102 723-10 - Intelligent Transport Systems (ITS); OSI cross-layer topics; Part 10: Interface between access layer and networking & transport layer*. V1.1.1. European Telecommunications Standards Institute. Nov. 2012.

[74] Michael Feiri, Jonathan Petit and Frank Kargl. 'Evaluation of Congestion-based Certificate Omission in VANETs'. In: *IEEE Vehicular Networking Conference (VNC)*. Nov. 2012, pp. 101–108. DOI: 10.1109/VNC.2012.6407417.

[75] Tom Henderson, Sally Floyd, Andrei Gurtov and Yoshifumi Nishida. *The NewReno Modification of TCP's Fast Recovery Algorithm*. RFC 6582. Apr. 2012. DOI: 10.17487/RFC6582.

[76] Bernhard Kloiber, Jérôme Härri and Thomas Strang. 'Dice the TX power — Improving Awareness Quality in VANETs by random transmit power selection'. In: *IEEE Vehicular Networking Conference (VNC)*. Nov. 2012, pp. 56–63. DOI: 10.1109/VNC.2012.6407445.

[77] Kathleen Nichols and Van Jacobson. 'Controlling Queue Delay'. In: *Queue* 10.5 (May 2012), pp. 20–34. DOI: 10.1145/2208917.2209336.

[78] Christoph Sommer, Stefan Jörer and Falko Dressler. 'On the Applicability of Two-Ray Path Loss Models for Vehicular Network Simulation'. In: *IEEE Vehicular Networking Conference (VNC)*. Nov. 2012, pp. 64–69. DOI: 10.1109/VNC.2012.6407446.

[79] Sundar Subramanian, Marc Werner, Shihuan Liu, Jubin Jose, Radu Lupoaie and Xinzhou Wu. 'Congestion Control for Vehicular Safety: Synchronous and Asynchronous MAC Algorithms'. In: *Proceedings of the Ninth ACM International Workshop on Vehicular Inter-networking, Systems, and Applications*. VANET '12. Low Wood Bay, Lake District, UK: ACM, 2012, pp. 63–72. ISBN: 978-1-4503-1317-9. DOI: 10.1145/2307888.2307900.

[80] RTCM Special Committee SC-104. *RTCM Standard 10403.2 - Differential GNSS (Global Navigation Satellite Systems) Services*. RTCM 10403.2. Radio Technical Commission for Maritime Services. Feb. 2013.

[81] Alessia Autolitano, Claudia Campolo, Antonella Molinaro, Riccardo M. Scopigno and Andrea Vesco. 'An Insight into Decentralized Congestion Control Techniques for VANETs from ETSI TS 102 687 V1.1.1'. In: *2013 IFIP Wireless Days (WD)*. Nov. 2013. DOI: 10.1109/WD.2013.6686471.

[82] Gaurav Bansal and John B. Kenney. 'Achieving weighted-fairness in message rate-based congestion control for DSRC systems'. In: *IEEE 5th International Symposium on Wireless Vehicular Communications (WiVeC)*. June 2013. DOI: 10.1109/wivec.2013.6698225.

[83] Gaurav Bansal, John. B. Kenney and Charles E. Rohrs. 'LIMERIC: A Linear Adaptive Message Rate Algorithm for DSRC Congestion Control'. In: *IEEE Transactions on Vehicular Technology* 62.9 (Nov. 2013), pp. 4182–4197. ISSN: 0018-9545. DOI: 10.1109/TVT.2013.2275014.

[84] Gaurav Bansal, Hongsheng Lu, John B. Kenney and Christian Poellabauer. 'EMBARC: Error Model Based Adaptive Rate Control for Vehicle-to-vehicle Communications'. In: *Proceeding of the Tenth ACM International Workshop on Vehicular Inter-networking, Systems, and Applications*. VANET '13. Taipei, Taiwan: ACM, 2013, pp. 41–50. ISBN: 978-1-4503-2073-3. DOI: 10.1145/2482967.2482972.

[85] David Eckhoff, Nikoletta Sofra and Reinhard German. 'A Performance Study of Cooperative Awareness in ETSI ITS G5 and IEEE WAVE'. In: *10th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*. Mar. 2013, pp. 196–200. DOI: 10.1109/WONS.2013.6578347.

[86] ETSI. *EN 302 663 - Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band*. V1.2.1. European Telecommunications Standards Institute. July 2013.

[87] ETSI. *TS 101 539-1 - Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification*. V1.1.1. European Telecommunications Standards Institute. Aug. 2013.

[88] ETSI. *TS 101 539-3 - Intelligent Transport Systems (ITS); V2X Applications; Part 3: Longitudinal Collision Risk Warning (LCRW) application requirements specification*. V1.1.1. European Telecommunications Standards Institute. Nov. 2013.

[89] ETSI. *TS 102 636-4-2 - Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 2: Media-dependent functionalities for ITS-G5*. V1.1.1. European Telecommunications Standards Institute. Oct. 2013.

[90] Thomas Hühn. 'A Measurement-Based Joint Power and Rate Controller for IEEE 802.11 Networks'. Doctoral Thesis. Berlin: Technische Universität Berlin, Fakultät IV - Elektrotechnik und Informatik, 2013. DOI: 10.14279/depositonce-3725. URL: http://dx.doi.org/10.14279/depositonce-3725.

[91] Hamid Menouar, Nestor Mariyasagayam, Massimiliano Lenardi and Yuki Horita. 'Method and apparartus for disseminating data in a communication network'. European pat. EP 2 592 869 A1. Hitachi Ltd., Tokyo, Japan. 15th May 2013.

[92] Michele Rondinone et al. 'iTETRIS: A modular simulation platform for the large scale evaluation of cooperative ITS applications'. In: *Simulation Modelling Practice and Theory* 34 (May 2013), pp. 99–125. DOI: 10.1016/j.simpat.2013.01.007.

[93] Tessa Tielert, Daniel Jiang, Hannes Hartenstein and Luca Delgrossi. 'Joint Power/Rate Congestion Control Optimizing Packet Reception in Vehicle Safety Communications'. In: *Proceeding of the Tenth ACM International Workshop on Vehicular Inter-networking, Systems, and Applications*. VANET '13. Taipei, Taiwan: ACM, 2013, pp. 51–60. ISBN: 978-1-4503-2073-3. DOI: 10.1145/2482967.2482968.

[94] Alessia Autolitano, Massimo Reineri, Riccardo M. Scopigno, Claudia Campolo and Antonella Molinaro. 'Understanding the Channel Busy Ratio Metrics for Decentralized Congestion Control in VANETs'. In: *International Conference on Connected Vehicles and Expo (ICCVE)*. Nov. 2014, pp. 717–722. DOI: 10.1109/ICCVE.2014.7297644.

[95] Gaurav Bansal, Bin Cheng, Ali Rostami, Katrin Sjöberg, John B. Kenney and Marco Gruteser. 'Comparing LIMERIC and DCC approaches for VANET channel congestion control'. In: *IEEE 6th International Symposium on Wireless Vehicular Communications (WiVeC 2014)*. Sept. 2014. DOI: 10.1109/WIVEC.2014.6953217.

[96] ETSI. *TR 101 612 - Intelligent Transport Systems (ITS); Cross Layer DCC Management Entity for operation in the ITS G5A and ITS G5B medium; Report on Cross layer DCC algorithms and performance evaluation*. Tech. rep. Sept. 2014.

[97] Abdelmajid Khelil and David Soldani. 'On the Suitability of Device-to-Device Communications for Road Traffic Safety'. In: *IEEE World Forum no Internet of Things (WF-IoT)*. Mar. 2014, pp. 224–229. DOI: 10.1109/WF-IoT.2014.6803163.

[98] European Parliament and Council. 'Directive 2014/53/EU on the harmonisation of the laws of the Member States relating to the making available on the market of radio equipment and repealing Directive 1999/5/EC'. In: *Official Journal of the European Union* 57.L 153 (May 2014), pp. 62–106. URL: http://eur-lex.europa.eu/eli/dir/2014/53/oj.

[99] Miguel Sepulcre, Javier Gozalvez, Onur Altintas and Haris Kremo. 'Adaptive Beaconing for Congestion and Awareness Control in Vehicular Networks'. In: *IEEE Vehicular Networking Conference (VNC)*. Dec. 2014, pp. 81–88. DOI: 10.1109/VNC.2014.7013313.

[100] Tessa Tielert. 'Rate-Adaptation Based Congestion Control for Vehicle Safety Communications'. PhD thesis. Karlsruhe: Karsruher Institut für Technologie (KIT), Fakultät für Informatik, 2014. DOI: 10.5445/IR/1000045971.

[101] *IETF Recommendations Regarding Active Queue Management*. RFC 7567. July 2015. DOI: 10.17487/RFC7567.

[102] Bin Cheng, Ali Rostami, Marco Gruteser, John B. Kenney, Gaurav Bansal and Katrin Sjöberg. 'Performance evaluation of a mixed vehicular network with CAM-DCC and LIMERIC vehicles'. In: *IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. June 2015. DOI: 10.1109/WoWMoM.2015.7158209.

[103] CEPT Electronic Communications Committee. *ECC Decision (08)01: The harmonised use of the 5875-5925 MHz frequency band for Intelligent Transport Systems (ITS)*. Appproved 14 March 2008, Amended 3 July 2015. July 2015.

[104] ETSI. *TR 101 613 - Intelligent Transport Systems (ITS); Cross Layer DCC Management Entity for operation in the ITS G5A and ITS G5B medium; Validation set-up and results*. Tech. rep. Sept. 2015.

[105] ETSI. *TS 102 792 - Intelligent Transport Systems (ITS); Mitigation techniques to avoid interference between European CEN Dedicated Short Range Communication (CEN DSRC) equipment and Intelligent Transport Systems (ITS) operating in the 5 GHz frequency range*. V1.2.1. European Telecommunications Standards Institute. June 2015.

[106] ETSI. *TS 103 097 - Intelligent Transport Systems (ITS); Security; Security header and certificate formats*. V1.2.1. European Telecommunications Standards Institute. June 2015.

[107] ETSI. *TS 103 175 - Intelligent Transport Systems (ITS); Cross Layer DCC Management Entity for operation in the ITS G5A and ITS G5B medium*. V1.1.1. European Telecommunications Standards Institute. June 2015.

[108] Toke Høiland-Jørgensen, Per Hurtig and Anna Brunstrom. 'The Good, the Bad and the WiFi: Modern AQMs in a residential setting'. In: *Computer Networks* 89 (2015), pp. 90–106. ISSN: 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2015.07.014.

[109] Sebastian Kühlmorgen, Ignacio Llatser, Andreas Festag and Gerhard Fettweis. 'Performance Evaluation of ETSI GeoNetworking for Vehicular Ad Hoc Networks'. In: *81st Vehicular Technology Conference (VTC Spring)*. May 2015. DOI: 10.1109/VTCSpring.2015.7146003.

[110] Thomas Paulin and Stefan Rührup. 'On the Impact of Fading and Interference on Contention-Based Geographic Routing in VANETs'. In: *82nd Vehicular Technology Conference (VTC2015-Fall)*. Sept. 2015. DOI: 10.1109/VTCFall.2015.7391044.

[111] Dieter Smely, Stefan Rührup, Robert K. Schmidt, John Kenney and Katrin Sjöberg. 'Decentralized Congestion Contorl Techniques for VANETs'. In: *Vehicle ad hoc Networks*. Ed. by Claudia Campolo, Antonella Molinaro and Riccardo Scopigno. Cham: Springer, 2015, pp. 165–191. ISBN: 978-3-319-15496-1. DOI: 10.1007/978-3-319-15497-8.

[112] Bengi Aygun, Mate Boban and Alexander M. Wyglinski. 'ECPR: Environment-and context-aware combined power and rate distributed congestion control for vehicular communications'. In: *Computer Communications* 93 (2016), pp. 3–16. DOI: 10.1016/j.comcom.2016.05.015.

[113] Mate Boban and Pedro M. d'Orey. 'Exploring the Practical Limits of Cooperative Awareness in Vehicular Communications'. In: *IEEE Transactions on Vehicular Technology* 65.6 (June 2016), pp. 3904–3916. DOI: 10.1109/TVT.2016.2544935.

[114] Martin A. Brown, Frederico Bolelli and Natale Particiello. *Traffic Control HOWTO*. Revision 1.1.0. The Linux Documentation Project, Mar. 2016. URL: http://tldp.org/en/Traffic-Control-HOWTO/index.html.

[115] David Eckhoff, Alexander Brummer and Christoph Sommer. 'On the Impact of Antenna Patterns on VANET simulation'. In: *IEEE Vehicular Networking Conference (VNC)*. Dec. 2016. DOI: 10.1109/VNC.2016.7835925.

[116] Arrate Alonso Gómez and Christoph F. Mecklenbräuker. 'Dependability of Decentralized Congestion Control for Varying VANET Density'. In: *IEEE Transactions on Vehicular Technology* 65.11 (Nov. 2016), pp. 9153–9167. ISSN: 0018-9545. DOI: 10.1109/TVT.2016.2519598.

[117] Bernhard Kloiber, Jérôme Härri Härri, Thomas Strang, Stephan Sand and Cristina Rico García. 'Random Transmit Power Control for DSRC and its Application to Cooperative Safety'. In: *IEEE Transactions on Dependable and Secure Computing* 13.1 (2016), pp. 18–31. DOI: 10.1109/TDSC.2015.2449845.

[118] Sven Laux, Gurjashan Singh Pannu, Stefan Schneider, Jan Tiemann, Florian Klingler, Christoph Sommer and Falko Dressler. 'OpenC2X - An Open Source Experimental and Prototyping Platform Supporting

ETSI ITS-G5'. In: *8th IEEE Vehicular Networking Conference (VNC 2016), Demo Session*. Columbus, OH: IEEE, Dec. 2016, pp. 152–153. DOI: 10.1109/VNC.2016.7835955. URL: http://www.ccs-labs.org/software/openc2x/ (visited on 15/07/2019).

[119] Ali Rostami, Bin Cheng, Gaurav Bansal, Katrin Sjöberg, Marco Gruteser and John B. Kenney. 'Stability Challenges and Enhancements for Vehicular Channel Congestion Control Approaches'. In: *IEEE Transactions on Intelligent Transportation Systems* 17.10 (2016), pp. 2935–2948. DOI: 10.1109/TITS.2016.2531048.

[120] SAE International. *J2945/1 - On-Board System Requirements for V2V Safety Communications*. Mar. 2016. DOI: https://doi.org/10.4271/J2945/1_201603.

[121] IEEE Computer Society. *Std 802.11-2016, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. Institute of Electrical and Electronics Engineers. 2016.

[122] Alexey Voronov, Jan De Jongh, Dennis Heuven and Albin Severinson. *Implementation of ETSI ITS G5 GeoNetworking stack, in Java: CAM-DENM / ASN.1 PER / BTP / GeoNetworking*. June 2016. DOI: 10.5281/zenodo.55650. URL: https://github.com/alexvoronov/geonetworking (visited on 15/07/2019).

[123] Alessandro Bazzi, Barbara M. Masini, Alberto Zanella and Ilaria Thibault. 'On the Performance of IEEE 802.11p and LTE-V2V for the Cooperative Awareness of Connected Vehicles'. In: *IEEE Transactions on Vehicular Technology* 66.11 (Nov. 2017), pp. 10419–10432. ISSN: 0018-9545. DOI: 10.1109/TVT.2017.2750803.

[124] Thiwiza Bellache, Oyunchimeg Shagdar and Samir Tohme. 'An Alternative Congestion Control using Enhanced Contention Based Forwarding for Vehicular Networks'. In: *13th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*. Feb. 2017, pp. 81–87. DOI: 10.1109/WONS.2017.7888763.

[125] Thiwiza Bellache, Oyunchimeg Shagdar and Samir Tohme. 'DCC-enabled contention based forwarding scheme for VANETs'. In: *13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. Oct. 2017. DOI: 10.1109/WiMOB.2017.8115767.

[126] Daniel-Maurice Burgstahler. 'Collaborative Sensing in Automotive Scenarios: Enhancements of the Vehicular Electronic Horizon through Collaboratively Sensed Knowledge'. PhD thesis. Darmstadt: Technische Universität, 2017. URL: http://tuprints.ulb.tu-darmstadt.de/6554/.

[127] Laura Codeca, Raphael Frank and Thomas Engel. 'Luxembourg SUMO Traffic (LuST) Scenario: Traffic Demand Evaluation'. In: *IEEE Intelligent Transportation Systems Magazine* 9.2 (2017), pp. 52–63. DOI: 10.1109/MITS.2017.2666585.

[128] ETSI. *EN 302 571 - Intelligent Transport Systems (ITS); Radiocommunications equipment operating in the 5855 to 5925 MHz frequency band; Harmonised Standard covering the essential requirements of article 3.2 of Directive 2014/53/EU*. V2.1.1. European Telecommunications Standards Institute. Feb. 2017.

[129] Laurent Gallo and Jérôme Härri. 'Unsupervised Long-Term Evolution Device-to-Device: A Case Study for Safety-Critical V2X Communications'. In: *IEEE Vehicular Technology Magazine* 12.2 (2017), pp. 69–77. DOI: 10.1109/10.1109/MVT.2017.2669346.

[130] Hendrik-Jörn Günther. 'Collective Perception in Vehicular Ad-hoc Networks'. PhD thesis. Braunschweig: Technische Universität Carolo-Wilhelmina, 2017. DOI: 10.24355/dbbs.084-201802021049. URL: https://publikationsserver.tu-braunschweig.de/receive/dbbs_mods_00065561.

[131] Toke Høiland-Jørgensen, Michał Kazior, Dave Täht and Per Hurtig. 'Ending the Anomaly: Achieving Low Latency and Airtime Fairness in WiFi'. In: *2017 USENIX Annual Technical Conference (USENIX ATC 17)*. USENIX Association, July 2017, pp. 139–151. URL: https://www.usenix.org/conference/atc17/technical-sessions/presentation/hoilan-jorgesen.

[132] Charles F. F. Karney. *GeographicLib*. Version 1.49. Oct. 2017. URL: https://geographiclib.sourceforge.io/1.49 (visited on 22/07/2019).

[133] Rong Pan, Preethi Natarajan, Fred Baker and Greg White. *Proportional Integral Controller Enhanced (PIE): A Lightweight Control Scheme to Address the Bufferbloat Problem*. RFC 8033. Feb. 2017. DOI: 10.17487/RFC8033.

[134] Greg White and Rong Pan. *Active Queue Management (AQM) Based on Proportional Integral Controller Enhanced (PIE) for Data-Over-Cable Service Interface Specifications (DOCSIS) Cable Modems*. RFC 8034. Feb. 2017. DOI: 10.17487/RFC8034.

[135] 3rd Generation Partnership Project (3GPP). *TS 122 185 - LTE; Service requirements for V2X services (3GPP TS 22.185)*. Version 15.0, Release 15. European Telecommunications Standards Institute. July 2018.

[136] 3rd Generation Partnership Project (3GPP). *TS 122 186 - 5G; Service requirements for enhanced V2X scenarios (3GPP TS 22.186)*. Version 15.4, Release 15. European Telecommunications Standards Institute. Oct. 2018.

[137] Pablo Alvarez Lopez et al. 'Microscopic Traffic Simulation using SUMO'. In: *Proceedings of the 21st IEEE International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, Nov. 2018, pp. 2575–2582. DOI: 10.1109/ITSC.2018.8569938.

[138] Boris Atanassow and Katrin Sjöberg. *Platooning protocol definition and Communication strategy. D2.8 of H2020 project ENSEMBLE*. Tech. rep. 2018. URL: https://platooningensemble.eu (visited on 07/07/2020).

[139] CAR 2 CAR Communication Consortium. *Basic System Profile*. Release 1.3.0. Aug. 2018.

[140] Lara Codeca and Jérôme Härri. 'Monaco SUMO Traffic (MoST) Scenario: A 3D Mobility Scenario for Cooperative ITS'. In: *SUMO User Conference, Simulating Autonomous and Intermodal Transport Systems*. Berlin, Germany, May 2018.

[141] Cohda Wireless. *Product brief sheet MK5 OBU*. 2018. URL: https://www.cohdawireless.com/wp-content/uploads/2018/08/CW_Product-Brief-sheet-MK5-OBU.pdf (visited on 08/08/2019).

[142] ETSI. *TS 101 539-2 - Intelligent Transport Systems (ITS); V2X Applications; Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification*. V1.1.1. European Telecommunications Standards Institute. June 2018.

[143] ETSI. *TS 102 687 - Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part*. V1.2.1. European Telecommunications Standards Institute. Apr. 2018.

[144] ETSI. *TS 103 301 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Facilities layer protocols and communication requirements for infrastructure services*. V1.2.1. European Telecommunications Standards Institute. Aug. 2018.

[145] Toke Hoeiland-Joergensen, Paul McKenney, Dave Taht, Jim Gettys and Eric Dumazet. *The Flow Queue CoDel Packet Scheduler and Active Queue Management Algorithm*. RFC 8290. Jan. 2018. DOI: 10.17487/RFC8290.

[146] ISO. *ISO 26262: Road vehicles - Functional safety*. International Organization for Standardization. Dec. 2018.

[147] Irfan Khan and Jérôme Härri. 'Flexible Packet Generation Control for Multi-Application V2V Communication'. In: *IEEE 88th Vehicular Technology Cnoference (VTC2018-Fall)*. Aug. 2018. DOI: 10.1109/VTCFall.2018.8690653.

[148] Irfan Khan and Jérôme Härri. 'Integration Challenges of Facilities-Layer DCC for Heterogeneous V2X Services'. In: *IEEE Intelligent Vehicles Symposium (IV)*. June 2018, pp. 1131–1136. DOI: 10.1109/IVS.2018.8500572.

[149] Sebastian Kühlmorgen, Patrick Schmager, Andreas Festag and Gerhard Fettweis. 'Simulation-Based Evaluation of ETSI ITS-G5 and Cellular-VCS in a Real-World Traffic Scenario'. In: *IEEE 8th Vehicular Technology Conference (VTC-Fall)*. Aug. 2018. DOI: 10.1109/VTCFall.2018.8691011.

[150] Tom H. Luan and Xuemin (Sherman) Shen. 'A Queueing Based Model for Analyzing Multihop Performance in VANET'. In: *IEEE International Conference on Communication Systems (ICCS)*. Dec. 2018, pp. 186–191. DOI: 10.1109/ICCS.2018.8689179.

[151] Vincent Martinez and Friedbert Berens. *Survey on ITS-G5 CAM statistics*. Tech. rep. Dec. 2018.

[152] Kathleen Nichols and Van Jacobson. *Controlled Delay Active Queue Management*. RFC 8289. Jan. 2018. DOI: 10.17487/RFC8289.

[153] Miguel Sepulcre, Pedro Tercero and Javier Gozalvez. 'Can Beacons be Compressed to Reduce the Channel Load in Vehicular Networks?' In: *2018 IEEE Vehicular Networking Conference (VNC)*. VNC 2018. Dec. 2018. DOI: 10.1109/VNC.2018.8628386.

[154] Angelo Trotta and Luca Sciullo. 'QoS-Based Mobility System for Autonomous Unmanned Aerial Vehicles Wireless Networks'. In: *Wired/Wireless Internet Communications*. Cham: Springer International Publishing, 2018, pp. 233–245. DOI: 10.1007/978-3-030-02931-9_19.

[155] 5GAA. *C-V2X Use Cases - Methodology, Examples and Service Level Requirements*. White Paper 1.0. June 2019.

[156] 5GAA. *Making 5G Proactive and Predictive for the Automotive Industry*. White Paper 1.0. Dec. 2019.

[157] Soheil Abbasloo and H. Jonathan Chao. 'Bounding Queue Delay in Cellular Networks to Support Ultra-Low Latency Applications'. In: *arXiv e-prints* arXiv:1908.00953 (Aug. 2019). under review.

[158] Daniel D. Adriano, Antonio G.N. Novaes and Michelle S. Wangham. 'Providing a Dynamic Milk-Run Vehicle Routing Using Vehicular Ad hoc Networks'. In: *Proceedings of the 9th ACM Symposium on Design and Analysis of Intelligent Vehicular Network and Applications*. DIVANet '19. Miami Beach, FL, USA: ACM, 2019, pp. 31–38. DOI: 10.1145/3345838.3356006.

[159] Christoph Allig and Gerd Wanielik. 'Extending the vehicular network simulator Artery in order to generate synthetic data for collective perception'. In: *Advances in Radio Science* 17 (2019), pp. 189–196. DOI: 10.5194/ars-17-189-2019.

[160] Alessandro Bazzi. 'Congestion Control Mechanisms in IEEE 802.11p and Sidelink C-V2X'. In: *53rd Asilomar Conference on Signals, Systems, and Computers*. 2019, pp. 1125–1130.

[161] Thomas Blazek, Golsa Ghiaasi, Christian Backfrieder, Gerald Ostermayer and Christoph F. Mecklenbräuker. 'Performance modelling and analysis for vehicle-to-anything connectivity in representative high-interference channels'. In: *IET Microwaves, Antennas Propagation* 13.13 (2019), pp. 2216–2223. DOI: 10.1049/iet-map.2018.6183.

[162] Bastian Bloessl and Aisling O'Driscoll. 'A Case for Good Defaults: Pitfalls in VANET Physical Layer Simulations'. In: June 2019. DOI: 10.1109/WD.2019.8734227.

[163] CAR 2 CAR Communication Consortium. *Basic System Profile*. Release 1.4.0. Sept. 2019.

[164] CAR 2 CAR Communication Consortium. *Guidance for day 2 and beyond roadmap*. Tech. rep. Sept. 2019.

[165] CAR 2 CAR Communication Consortium. *Triggering Conditions and Data Quality - Adverse Weather Conditions*. Release 1.4.0. Sept. 2019.

[166] CAR 2 CAR Communication Consortium. *Triggering Conditions and Data Quality - Dangerous Situation*. Release 1.4.0. Sept. 2019.

[167] CAR 2 CAR Communication Consortium. *Triggering Conditions and Data Quality - Exchange of IRCs*. Release 1.4.0. Sept. 2019.

[168] CAR 2 CAR Communication Consortium. *Triggering Conditions and Data Quality - Pre-Crash Information*. Release 1.4.0. Sept. 2019.

[169] CAR 2 CAR Communication Consortium. *Triggering Conditions and Data Quality - Traffic Jam*. Release 1.4.0. Sept. 2019.

[170] ETSI. *EN 302 636-5-1 - Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol*. V2.2.1. European Telecommunications Standards Institute. May 2019.

[171] ETSI. *EN 302 637-2 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*. V1.4.1. European Telecommunications Standards Institute. Apr. 2019.

[172] ETSI. *EN 302 637-3 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service*. V1.3.1. European Telecommunications Standards Institute. Apr. 2019.

[173] ETSI. *TR 103 257-1 - Intelligent Transport Systems (ITS); Access Layer; Part 1: Channel Models for the 5.9 GHz frequency band*. Tech. rep. May 2019.

[174] ETSI. *TR 103 299 - Intelligent Transport Systems (ITS); Cooperative Adaptive Cruise Control (CACC); Pre-standardization study*. Tech. rep. June 2019.

[175] ETSI. *TR 103 562 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS); Release 2*. Tech. rep. Dec. 2019.

[176] ETSI. *TS 103 248 - Intelligent Transport Systems (ITS); GeoNetworking; Port Numbers for the Basic Transport Protocol (BTP)*. V1.3.1. European Telecommunications Standards Institute. Apr. 2019.

[177] European Commission. *Commission Delegated Regulation supplementing Directive 2010/40/EU of the European Parliament and of the Council with regard to the deployment and operational use of cooperative intelligent transport systems*. C(2019) 1789 final. 2019.

[178] Foad Hajiaghajani and Chunming Qiao. 'IEEE Global Communications Conference (GLOBECOM)'. In: *Tailgating Risk-Aware Beacon Rate Adaptation for Distributed Congestion Control in VANETs*. Dec. 2019. DOI: 10.1109/GLOBECOM38437.2019.9013608.

[179] Mohammad Irfan Khan, Stefanie Sesia and Jérôme Härri. 'In Vehicle Resource Orchestration for Multi-V2X Services'. In: *IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*. Sept. 2019. DOI: 10.1109/VTCFall.2019.8891527.

[180] Valerian Mannoni, Vincent Berg, Stefania Sesia and Eric Perraud. 'A Comparison of the V2X Communication Systems: ITS-G5 and C-V2X'. In: *89th Vehicular Technology Conference (VTC)*. Apr. 2019. DOI: 10.1109/VTCSpring.2019.8746562.

[181] Tobias Renzler, Michael Stolz and Daniel Watzenig. 'Decentralized Dynamic Platooning Architecture with V2V Communication Tested in Omnet++'. In: *IEEE International Conference on Connected Vehicles and Expo (ICCVE)*. Nov. 2019. DOI: 10.1109/ICCVE45908.2019.8965224.

[182] João Rufino, Luis Silva, Bruno Fernandes, João Almeida and Joaquim Ferreira. 'Overhead of V2X secured messages: an analysis'. In: *89th Vehicular Technology Conference (VTC-Spring)*. Apr. 2019. DOI: 10.1109/VTCSpring.2019.8746479.

[183] Ignacio Soto, Oscar Amador, Manuel Urueña and Maria Calderon. 'Strengths and Weaknesses of the ETSI Adaptive DCC Algorithm: A Proposal for Improvement'. In: *IEEE Communications Letters* 23.5 (May 2019), pp. 802–805. DOI: 10.1109/LCOMM.2019.2906178.

[184] Bruno Vieira, Ricardo Severino, Anis Koubaa and Eduardo Tovar. 'Towards a Realisitic Simulation Framework for Vehicular Platooning Applications'. In: *22nd International Symposium on Real-Time Distributed Computing (ISORC)*. May 2019. DOI: 10.1109/ISORC.2019.00028.

[185] Philip Wendland, Guenter Schaefer and Reiner S. Thomä. 'LTE-V2X Mode 4: Increasing Robustness and DCC Compatibility with Reservation Splitting'. In: *IEEE International Conference on Connected Vehicles and Expo (ICCVE)*. Nov. 2019. DOI: 10.1109/ICCVE45908.2019.8965212.

[186] Sijie Zhu, Dip Goswami and Hong Li. 'Evaluation Platform of Platoon Control Algorithms in Complex Communication Scenarios'. In: Apr. 2019. DOI: 10.1109/VTCSpring.2019.8746477.

[187] 3rd Generation Partnership Project (3GPP). *TS 123 501 - 5G; System architecture for the 5G System (5GS) (3GPP TS 23.501)*. Version 15.8, Release 15. European Telecommunications Standards Institute. Jan. 2020.

[188] 5GAA. *TR-200055 - 5GS Enhancements for Providing Predictive QoS in C-V2X*. Tech. rep. Feb. 2020.

[189] Thales Teixeira de Almeida, Lucas Carvalho de Gomes, Fernando Molano Ortiz, José Geraldo Ribeiro Júnior and Luís Henrique M. K. Costa. 'Comparative Analysis of a Vehicular Safety Application in NS-3 and Veins'. In: *IEEE Transactions on Intelligent Transportation Systems* Early Access (2020). DOI: 10.1109/TITS.2020.3014840.

[190] Oscar Amador, Ignacio Soto, Calderón Maria and Manuel Urueña. 'Experimental Evaluation of the ETSI DCC Adaptive Approach and Related Algorithms'. In: *IEEE Access* 8 (2020), pp. 49798–49811. DOI: 10.1109/ACCESS.2020.2980377.

[191] ASFINAG. *ASFINAG connects roads with vehicles on a large scale in Europe*. Press Release. Oct. 2020. URL: https://www.c-roads.eu/fileadmin/user_upload/pics/News/PA_ASFINAG_Connects_Roads_with_Vehicles.pdf (visited on 30/11/2020).

[192] Autotalks. *Functional Safety for Enabling Present and Future V2X Use-Cases*. White Paper. Apr. 2020. URL: https://www.auto-talks.com/wp-content/uploads/2020/04/Functional-Safety-for-V2X-use-cases.pdf.

[193] ETSI. *EN 302 636-4-1 - Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 1: Media-Independent Functionality*. V1.4.1. European Telecommunications Standards Institute. Jan. 2020.

[194] ETSI. *TR 101 607 - Intelligent Transport Systems (ITS); Cooperative ITS (C-ITS); Release 1*. Tech. rep. Feb. 2020.

[195] ETSI. *TS 102 636-4-2 - Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 2: Media-dependent functionalities for ITS-G5*. V1.3.1. European Telecommunications Standards Institute. Aug. 2020.

[196] ETSI. *TS 102 636-4-3 - Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 3: Media-dependent functionalities for LTE-V2X*. V1.1.1. European Telecommunications Standards Institute. Aug. 2020.

[197] ETSI. *TS 102 965- Intelligent Transport Systems (ITS); Application Object Identifier (ITS-AID); Registration*. V1.5.1. European Telecommunications Standards Institute. Jan. 2020.

[198] ETSI. *TS 103 141 - Intelligent Transport Systems (ITS); Facilities Layer; DCC Entity for ITS-G5 Access Technology*. V0.1.1. European Telecommunications Standards Institute. May 2020.

[199] Mouna Karoui, Gerard Chalhoub and Antonio Freitas. 'A study of congestion control approaches for vehicular communications using ITS-G5'. In: *IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*. Aug. 2020. DOI: 10.1109/PIMRC48278.2020.9217143.

[200] Sebastian Kühlmorgen, Hongsheng Lu, Andreas Festag, John Kenney, Sebastian Gemsheim and Gerhard Fettweis. 'Evaluation of Congestion-Enabled Forwarding With Mixed Data Traffic in Vehicular Communications'. In: *IEEE Transactions on Intelligent Transportation Systems* 21.1 (Jan. 2020). DOI: 10.1109/TITS.2018.2890619.

[201] Nikita Lyamin, Boris Bellalta and Alexey Vinel. 'Age-of-Information-Aware Decentralized Congestion Control in VANETs'. In: 2.1 (2020), pp. 33–37. DOI: 10.1109/LNET.2020.2970695.

[202] Gokulnath Thandavarayan, Miguel Sepulcre and Javier Gozalvez. 'Cooperative Perception for Connected and Automated Vehicles: Evaluation and Impact of Congestion Control'. In: *IEEE Access* (2020). Under review.

[203] Volkswagen Communications. *Technical milestone in road safety: experts praise Volkswagen's Car2X technology*. Press Release. Mar. 2020. URL: https://www.volkswagen-newsroom.com/en/press-release/technical-milestone-in-road-safety-experts-praise-volkswagens-car2x-technology-5914 (visited on 30/11/2020).

[204] Yeomyung Yoon and Hyogon Kim. 'Balancing Power and Rate Control for Improved Congestion Control in Cellular V2X Communication Environments'. In: *IEEE Access* 8 (2020), pp. 105071–105081. DOI: 10.1109/ACCESS.2020.2999925.

[205] Johannes Berg, Valent Turkovic, Stuart Cheshire et al. *Linux Wireless: en:developers:documentation:mac80211:ratecontrol:minstrel*. Last modified on 2016-01-12. URL: https://wireless.wiki.kernel.org/en/developers/documentation/mac80211/ratecontrol/minstrel (visited on 17/02/2020).

[206] Federal Communications Commission. *FCC Modernizes 5.9 GHz Band For Wi-Fi and Auto Safety*. URL: https://docs.fcc.gov/public/attachments/DOC-368228A1.pdf (visited on 28/11/2020).

[207] OpenStreetMap contributors. Available under the Open Database License. URL: https://www.openstreetmap.org (visited on 05/11/2019).

[208] OpenStreetMap Wiki contributors. *Map Features*. Page Version ID: 1879758. URL: https://wiki.openstreetmap.org/wiki/Map_Features (visited on 30/10/2019).

[209] Marc Torrent-Moreno, Steven Corroy, Felix Schmidt-Eisenlohr and Hannes Hartenstein. 'IEEE 802.11-based One-Hop Broadcast Communications: Understanding Transmission Success and Failure under Different Radio Propagation Environments'. In: *Proceedings of the 9th ACM International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems*. MSWiM '06. Terromolinos, Spain: ACM, pp. 68–77. DOI: 10.1145/1164717.1164731.

[210] Lev Walkin. *asn1c - The ASN.1 Compiler*. URL: https://github.com/vlm/asn1c (visited on 22/07/2019).